Grounded Dual-Stream World Models at Scale: The Perception Stream Stays Load-Bearing

Justin Hill

Independent Research

justin@epagoge.io · Follow-up report · 21 July 2026

A follow-up to the July 2026 preprint "Grounded Dual-Stream Language Models" (Hill). It carries the founding perception ablation across three larger from-scratch runs and continues the lineage to 1.939B. Dated 2026-07-21.

Abstract

This paper follows the July 2026 preprint *Grounded Dual-Stream Language Models* (Hill), and reports what four larger runs over roughly two weeks did to its founding result, together with a run-integrity correction on the largest. The architecture is an original grounded dual-stream world model, built and trained from scratch. The founding ablation at 4.79M parameters fixes the object of study: with weights held fixed, removing the perception stream drops held-out authored-token accuracy from 0.9503 to 0.3756. Four control layers sit on the grounded objective. An autopilot adjusts the grounded-step fraction *p_struct* multiplicatively against explicit KPI checkpoints; ATLAS steers which grounded families sample; LOOM reaches subtypes below the family level; and two interpretability instruments, WARP at consolidation time and WEFT at formation time, read model internals so that grounded data is spent on genuine gaps rather than routing faults. The ablation collapse persists at every scale, staying between +0.44 and +0.60 and never falling back toward chance: +0.444 at the 526M continued pretraining, and +0.603 at 124M after a knowledge injection that left it marginally more causal rather than less. The two continued-pretrains leave inverse weight rel-L2 signatures, one ordering the mirror of the other. The most consequential defects found across this lineage were in the steering instruments, not the model. The 1.939B run, launched from scratch on the venue corpus lineage and halted at step 7133, we report as diagnostic-only. It trains cleanly at this scale (bits per byte 3.517 to 1.114, GHLE authored-token accuracy 0.265 to 0.988, zero NaN across the run) and it carries no capability claim, because its curriculum controller steered on a drive that was roughly 78% phantom need, its tag-level curriculum was silently inert across the product half of the corpus, its stream mixture was changed mid-run without a frozen manifest, and its two internal grounding readouts were built but never wired into the loop. Autonomous tool-call emission reads 0.00 at every checkpoint, which we diagnose as pre-liftoff and not model failure: the reading follows from the teacher-forced versus free-generation distinction, and a 124M positive control reads 1.0 on the same probe. We treat the run as uncontrolled rather than failed, and the correct response is a controlled re-run under a frozen configuration and a versioned mixture manifest.

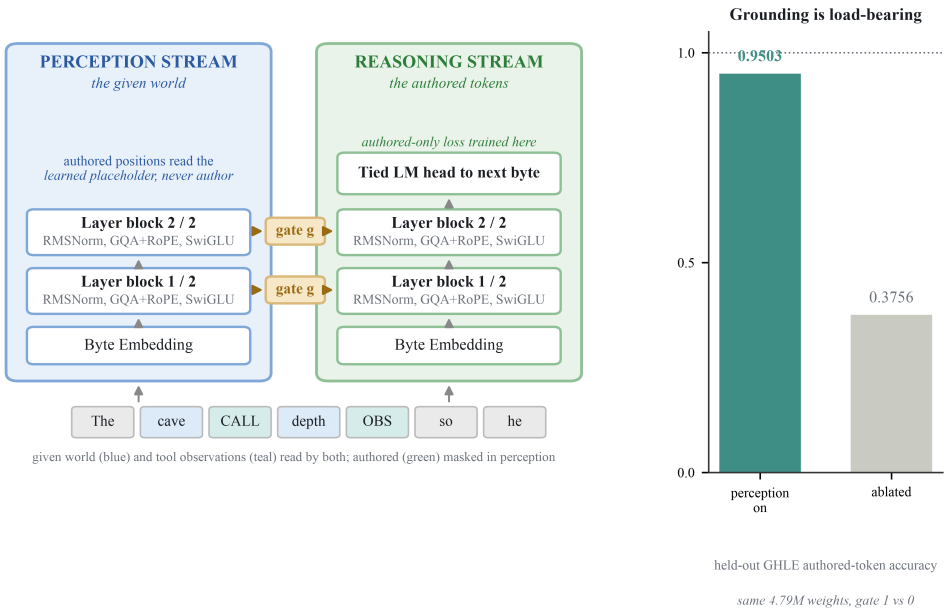

Figure 1. The grounded dual-stream architecture and its founding result. A perception stream reads the given world, the prompt together with verified tool observations, with the model's own authored positions replaced by a learned placeholder so that stream never authors. A reasoning stream generates and cross-attends to perception causally, gated per example so that ordinary text pretraining is the case with the gate off. At fixed 4.79M weights on 58 held-out rows, ablating the perception stream drops held-out authored-token accuracy from 0.9503 to 0.3756.

1. The founding gap and the lineage

This paper is a follow-up. In July 2026 the preprint "Grounded Dual-Stream Language Models" reported one mechanistic result at 4.79M parameters and explicitly deferred the question of capability at scale. What follows measures that founding gap against larger runs, across a two-week arc. On 2026-07-17 and 07-18 a 526M continued pretraining ran to step 800,000 with zero NaN. On 2026-07-19 a 124M pair reached its knowledge-injection phase under the M2 fair meters. On 2026-07-20 a 1.939B run launched from scratch on the grounded spine plus the venue corpus. On 2026-07-21, today, that run is halted at step 7133 and reported as diagnostic-only, for reasons Section 6 sets out in full. A lineage table records the configurations, parameter counts, ablation gaps, and standings; the prose keeps to what the runs were for.

Run	Params	Dates	Δ	Status
Preprint (v1)	4.79M	Jul 2026	+0.575	founding
526M CPT	526M	Jul 17-18	+0.444	replicated
124M pair	124M	Jul 19	+0.603	replicated
1.939B run	1.94B	Jul 20-21	n/a	diagnostic

Table 1. The lineage measured in this report. The ablation gap is the fixed-weight drop in held-out authored-token accuracy when the perception stream is removed. The 1.939B run carries no gap because it is reported as diagnostic-only (Section 6).

Everything downstream rests on one measurement. At 4.79M parameters, on 58 held-out GHLE rows, ablating the perception stream at fixed weights drops held-out authored-token accuracy from 0.9503 to 0.3756. That is $\Delta = 0.5747$ absolute, with no weights touched: the same trained model, the perception stream zeroed at inference. Bits per byte over the same rows splits the same way, 1.5239 with the read intact against 2.7513 with it severed. The perception stream reads the given world (the prompt plus masked tool observations, authored positions replaced by a learned placeholder token) and never authors a byte; the reasoning stream authors output and cross-attends to it causally, *is_causal* true so no future observation leaks. Remove the read, and the author loses more than half of its held-out accuracy.

The number is stated this way because it is falsifiable. Suppose grounding were a decoration, a scaffold the reasoning stream learns to route around once it has enough capacity to memorize the authored distribution outright. Then the gap has a prediction attached to it: it should shrink with scale, because a larger model can carry the answers in its weights and stop consulting the read. The scaled runs were built to check that prediction, and to force it to fail loudly if it was going to fail.

It did not vanish. At 526M the same fixed-weight ablation read g=1 0.9898 against g=0 0.5455, a gap of +0.444, still most of the authored accuracy. At 124M the gap was +0.575, and after a 29 GB web knowledge injection it stood at +0.603, with g=1 authored accuracy essentially unmoved at 0.9904. The collapse stays large at every scale, and injecting knowledge did not let the reasoning stream route around perception.

That result carried four assumptions forward, and each earned its place by surviving. The first is that the collapse is a mechanism rather than a small-scale artifact: the drop is what the architecture does when its input channel is severed, at 4.79M and again at every larger scale. The second is a claim about where factual truth should live. It belongs in the tool and observation channel that the perception stream reads rather than in the weights, because a model that reaches for a value through the read cannot silently substitute a memorized approximation for it. The third separates two things that are easy to conflate: knowledge and grounding occupy distinct machinery, so injecting facts should not, by itself, move the grounding signature, and the ablation gap should hold whether or not the model has been fed more to remember. The fourth is economic. A grounding objective is expensive, and a static data mixture wastes it, spending gradient steps on families that have already saturated while families that are still stalled go hungry. That last assumption is the one the entire control stack in Section 3 exists to act on.

None of these were free assertions at the 4.79M scale. They were bets, made on a single founding gap, that the gap meant what it appeared to mean. Sections 2 through 7 are the record of holding those bets against three larger runs, each measured under its own probe stack. Some bets held. In other places the instruments turned out to be lying instead, and Sections 2 and 4 sort out which readings to trust. For the 1.939B run, Section 6 reports why a numerically clean training curve is entered as evidence that the architecture trains at scale and as nothing more.

2. Meter provenance

A capability number carries no meaning without the probe that produced it. Change the probe and the same checkpoint reports a different quantity, not a cleaner reading of the same one. Three probe generations sit across this lineage. Every figure in this paper is stamped with the era it came from, and nothing is compared across a boundary between eras. Table 2 records which meter graded which run.

M0 ran from the 526M start to roughly step 217k. The *kpi_probes* decoded at gate-0 (*g*=0, plain single-stream text) under a 384-byte context cap. Three of them, *toolcall*, *decision*, and *honest_stop*, sat pinned at 0.0. The pin was structural: at 384 bytes the trace was truncated before the CALL boundary, and the grader scored ungrounded *g*=0 logits. The model's actual competence never entered the number. That instrument is D1, the emission false floor, and Section 4 treats it in full.

M1 covers the tail of the 526M run, from about step 217k to its end, and the whole of 124M phase 1. The probes were moved to gate-1 (*g*=1, grounded dual-stream) decode, context was raised to 2048, and CALL detection was made dual-dialect. Two graders stayed soft. *lang_score* ran against a 40-word fallback list, because the pods carry no */usr/share/dict/words*, so most real words fell outside its view. *know_score* summed continuation log-probability, which rewards length.

M2 begins with 124M phase 2 on 2026-07-19 and continues to the present, the 2B included. Here *lang_score* falls back to the real corpus vocabulary, *know_score* uses per-byte mean log-probability, and reference slices are capped at 64 MB. These are the current fair meters. When this paper reports a language or knowledge figure without further qualification, it is an M2 figure.

One earlier claim does not survive its own meter. The statement that language was flat at 526M was an M1 fallback-list artifact: a 40-word list cannot see most of what the model wrote, so the model looked worse than it was. Graded against real vocabulary, the same checkpoint returns a 0.97 real-word fraction. The claim is withdrawn.

Not every measurement passes through the *kpi_probes*. Weight-change and activation-drift numbers are read straight off the checkpoint tensors, so they hold identically under M0, M1, and M2; the meter question does not touch them. The founding ablation belongs in the same category. Held-out authored-token accuracy under a fixed-weight perception knockout is a direct teacher-forced read on the authored positions, not one of the fragile probes, and at 4.79M parameters it drops from 0.9503 to 0.3756 regardless of which era did the grading.

$$\Delta = a_{g=1} - a_{g=0}, \qquad \Delta_{4.79\text{M}} = 0.9503 - 0.3756 = 0.5747$$

Era	Window	Probe state
M0	526M start to 217k	Gate-off decode with a 384-byte context cap. Emission, decision, and honest-stop pinned at zero regardless of the model.
M1	526M 217k to 124M ph.1	Fixed for gate-on decode at 2048-byte context. Fluency still on a 40-word fallback list; knowledge length-biased by summed log-probability.
M2	124M ph.2 to present	Real-vocabulary fluency fallback, per-byte-mean knowledge, capped reference slices. The current fair meters, used for the 1.9B run.

Table 2. The three meter eras. A number is read only against its era.

3. The control stack

Four layers sit between the corpus and the optimizer, and they share one reading of the training problem: a static mixture wastes a grounded objective, because it keeps spending compute on families that are already saturated while stalled families never get fed. The autopilot sets how much of each step is grounded. *ATLAS* sets which grounded families are sampled. *LOOM* reaches below the family to the tag. Two interpretability instruments, *WARP* and *WEFT*, read the model internals so that the grounded budget lands on a genuine gap and not on a routing fault. All four read the same signal, the canary.

3.1 The autopilot

The autopilot controls one scalar: p_struct, the probability that a given step draws a grounded dual-stream example (g=1) rather than plain single-stream text (g=0). It does not consult the loss. It reads the canary, where every record carries a set of KPI probes, and each probe k has a target τ_k. The controller holds an exponential moving average per KPI and moves the mixture against the remaining gap.

The update, per canary and per KPI k:

ema_k ← (1−ρ)·ema_k + ρ·kpi_k need_k = max(0, τ_k − ema_k) drive = Σ_{k in struct} need_k − Σ_{k in text} need_k p_struct ← clip(p_struct·exp(η·drive), p_lo, p_hi)

$$m_k \leftarrow (1-\rho)\, m_k + \rho\, \kappa_k, \qquad \text{need}_k = \max(0,\ \tau_k - m_k)$$

$$d = \sum_{k \in S} \text{need}_k - \sum_{k \in \mathcal{T}} \text{need}_k$$

$$p \leftarrow \text{clip}\left(p\, e^{\eta d},\ p_{\text{lo}},\ p_{\text{hi}}\right)$$

with ρ=0.35, η=1.2, p_lo=0.25, p_hi=0.55. ema_k is the smoothed reading of KPI k; need_k is how far that reading still sits below its target, floored at zero; drive is the signed pressure, the summed need of the struct KPIs minus the summed need of the text KPIs; p_struct is the grounded fraction, moved multiplicatively by exp(η·drive) and clipped to the band [0.25, 0.55]. The struct KPIs are grounding, toolcall, decision, honest_stop, and echo. The text KPIs are coherence, lang, know, and narrative. Positive drive raises the grounded fraction, negative drive lowers it.

A KPI is marked DONE once it reads at or above τ_k for streak_n=3 consecutive canaries, and a DONE KPI drops out of the sums, so the controller stops chasing a target it already holds. When every KPI is DONE the run is converged and ends. One rule overrides the arithmetic: if the grounding ema falls below 0.95, a floor breach forces p_struct ← max(p_struct, 0.55). The grounded signal cannot be traded away to buy a text metric.

At the 2B halt (step 7133, M2 telemetry, read off the pre-fix controller) p_struct sat pinned at the ceiling 0.55 with drive 2.83. Of that drive roughly 0.85 came from the genuinely starved toolcall KPI. The remaining ~2.2 came from three KPIs that no probe on the 2B ever produced, and section 4 returns to why the arithmetic read them as maximally starved. The grounding floor was never breached.

3.2 ATLAS

ATLAS carries per-family sampling weights. Each canary scores every family on an eval slice of about 40 held-out rows, and ATLAS nudges weight toward the families sitting furthest below their goal. It also tracks family velocity from consecutive-canary deltas, so a family that is already climbing under its current weight is not over-fed while a flat one waits. The autopilot decides how grounded the step is; ATLAS decides, within the grounded portion, where the mass goes.

3.3 LOOM

Below the family is the tag, and LOOM works four stages there. At load it carves a per-tag held-out slice. It scores each tag against that slice. Once a family weight is already at ceiling and a tag is still flat, it escalates that tag to undiluted inclusion. It then draws from the escalated pool. The generators are closures over the carved slices with no pre-registration, so an old family can steer one of its subtypes without a corpus rebuild.

LOOM depends on a tag field existing. When the venue families were registered without an oracle or tag field, LOOM had nothing to carve across the product half of the corpus and quietly went dark there. That is D3, and it is a finding about the instrument, treated in section 4.

3.4 The canary

One JSON record every 30 minutes.

That record carries per-family teacher-forced accuracy, the KPI probes, held-out bits per byte, and a frozen battery, and it is the single signal the autopilot, ATLAS, and LOOM all read. Because all three consume the same record, their decisions are commensurable: the family accuracy that moves an ATLAS weight is the same number the autopilot rolls into drive.

3.5 WARP and WEFT

The two names are literal. WARP reads late, along the direction of consolidation. WEFT reads early, across formation. Both exist so that grounded data is spent on a real gap rather than on a value that forms correctly and then fails to route.

WARP reads at consolidation time, on gradient alignment. It builds a per-family by per-concept occupancy fingerprint C[f,g] and

triages on it, so that only the ambiguous families pay the expensive 8B-LoRA gradient pass. name_donor(C,g) names the family to up-weight so it can donate a concept that is missing elsewhere.

WEFT reads at formation time, at the PREDICT token, before the masked <<<OBS returns. Its spine is FORM × USE. FORM asks whether the correct grounded value is poised in the workspace, read by a linear probe on the reasoning residual at the CALL/OBS boundary. USE asks whether that poised value routes into the sim-verified output, read as a nonzero ablation-collapse. The two binary reads cross into four verdicts:

$$\delta = v(x) - v(x \ominus u_g); \qquad \text{misrouted if } \delta > \delta_{\text{rand}}$$

HEALTHY (form ok, use ok). FORMS-BUT-MISROUTED (form ok, use fail): the lever is a routing or motor fix, ATLAS holds the mix, and adding data here is dilution. FAILS-TO-FORM (form fail, use fail): the lever is DATA, and ATLAS up-weights the family. SHORTCUT (form fail, verdict green, no ablation-collapse): brittle, and the loss-lies guard trips.

Never an aux-loss.

Optimizing the probe would train the model to satisfy the probe, which destroys the one signal worth having. The ambiguous cell between FORMS-BUT-MISROUTED and FAILS-TO-FORM is adjudicated interventionally instead: project the FORM direction out mid-forward and watch the sim verdict against a norm-matched random-direction control. Nonzero collapse means the value was real and misrouted, so the route lever applies; a null result means the direction was spurious, and the cell is reclassified to the data lever.

WEFT is built on the verified J-lens core, self-tested, and run against the 524M as a fit-and-ratify probe (declination_deg, solar corpus, layer-6 reasoning band). It is not yet wired as a live in-loop callback on the 2B. WARP is not yet live on the 2B either.

3.6 Admission

Admission governs whether the corpus is allowed to grow at all. On a plateau the corpus may expand from a pre-approved registry, but only after a fork diagnoses which kind of plateau it is. A capacity plateau (KPI flat, bpb still falling, weights still moving) means data cannot fix the problem, and expansion is forbidden. An exhaustion plateau (KPI flat, and bpb flattening, and per-family gradients shrinking) is the trigger. Meter saturation calls for new gates, not new data.

The diagnosis is itself a fork: branch a short probe, feed it the candidate stream, and measure velocity. Velocity where the old corpus produced none means admit. Nothing moving means do not spend the tokens. An admitted stream enters as a new ATLAS family carrying a prior-fitted forgetting constant, guarded by replay floors on the standing families, the same 12% replay mechanism the continued pretrains already leaned on.

4. Where the instruments failed

Every number in the preceding sections rode on an instrument, and four of those instruments were wrong for a stretch of the lineage before they were caught. None of the four touched the model weights, which is why they are reported here as findings about instruments rather than about the model: a defect in an instrument reads back as a defect in the thing being measured, and for a while each of these did exactly that. The costliest, D2, distorted the steering itself and shaped the archived 2B telemetry, so it is treated at length below. The other three cost less and are simpler to state.

4.1 D1, the emission false floor (M0)

Under the M0 meter the *toolcall*, *decision*, and *honest_stop* probes decoded at gate-0 with a 384-byte context cap. The cap truncated the trace before the *>>>CALL* boundary, so the probe graded ungrounded logits and returned 0.0 whatever the model had learned. All three struct KPIs read a hard zero for structural reasons, independent of what the model had actually emitted. The repair landed at M1: gate-1 decode, context raised to 2048, dual-dialect CALL detection. With that in place the 526M true *toolcall* surfaced at roughly 0.958, visible only after step ~217k. Before that step the meter reported a floor that belonged to the meter.

4.2 D2, phantom-need

This is the consequential one.

The autopilot computes per-canary need as $need\_k = \max(0, \tau\_k - ema\_k)$, where $ema\_k \leftarrow (1-\rho)\cdot ema\_k + \rho\cdot kpi\_k$ with $\rho=0.35$ is the exponential moving average of the k-th reading and τ_k its target. The grounded fraction then moves by $drive = \Sigma\_\{k \text{ in struct}\}\ need\_k - \Sigma\_\{k \text{ in text}\}\ need\_k$, followed by $p\_struct \leftarrow \text{clip}(p\_struct\cdot\exp(\eta\cdot drive), p\_lo, p\_hi)$ with $\eta=1.2$, $p\_lo=0.25$, $p\_hi=0.55$. The defect sat in the first sum. Need was accumulated over all target struct KPIs, *decision*, *honest_stop*, and *echo* among them, three KPIs that no probe on the 2B ever produced a reading for. An unmeasured KPI defaulted to *ema* 0.0, which the max inside *need_k* reads as maximally starved. Three KPIs that were never measured therefore each contributed a full τ_k of need on every canary.

At the 2B halt the arithmetic resolved to *p_struct* pinned at 0.55 and *drive* 2.83. Of that 2.83, roughly 0.85 was the genuinely-starved measured *toolcall* and roughly 2.2 was phantom, the three unmeasured KPIs summing a deficit that had no reading behind it. The controller was holding the grounded fraction at its ceiling to chase a shortfall that existed only in the absence of a probe.

The fix tracks a *measured* set and computes need only over measured, not-done targets, so an unmeasured canary no longer enters the sum. It passes its unit tests, P1 through P5. It has not been run. The archived 2B telemetry reported in Section 6 comes entirely from the pre-fix controller, and the *drive* value of 2.83 should be read as the pre-fix number that it is.

4.3 D3, lost tag granularity

LOOM carves a per-tag held-out slice at load and escalates a flat tag to undiluted inclusion once its family weight reaches ceiling. Neither step is possible without a tag. The venue families were registered without an oracle/tag field, so LOOM had nothing to carve and nothing to escalate across the product half of the corpus. It failed silent: no error surfaced and no escalation fired, the mechanism simply idle for those families. The repair re-tagged 13 venue families plus deep_full through an oracle-field tagger, restoring the field LOOM keys on.

4.4 D4, the OOM trap

A gate-1 struct forward at full context runs about 2.5x the memory of the gate-0 text branch. The launcher preflighted the cheap branch, so *struct_batch* 6 cleared the check and then OOM'd once it hit full context; *struct_batch* 3 did the same. What survived was *struct_batch* 2 with expandable_segments. The rule the trap leaves behind is short: preflight the worst branch, not the first.

5. The adaptations these forced

The defects of Section 4 changed what the pipeline measures and what it feeds. The first change concerns *know*. Across the 526M CPT, lm-eval placed the model well below Qwen2.5-0.5B on every standard slice: lambada 0.074 against 0.518, hellaswag_norm 0.372 against 0.504, arc_easy_norm 0.356 against 0.586, arc_challenge_norm 0.236 against 0.330, piqa_norm 0.538 against 0.710, winogrande 0.498 against 0.572. A byte-level transformer trained from scratch on a grounded spine does not carry the memorized world-knowledge that a subword model of comparable size absorbs from web text. The 124M phase-2 run confirmed the ceiling from the other side: under the M2 meter, *know* rose from a fair baseline of 0.438 to 0.562 and then plateaued while bits per byte kept descending.

So *know* was demoted. It stays on the canary as a tracked reading. It no longer enters the autopilot drive as a gating text KPI, which means its need_k can no longer pull *p_struct*.

Behind the demotion is a routing decision. Factual truth is handled through the tool and observation channel. The weights are not asked to carry it. The masked *<<<OBS* return already delivers the sim-verified value the reasoning stream needs, and the grounded objective rewards using that value rather than reciting it. Demoting *know* as a gate keeps the founding architecture honest: the perception stream reads the given world, the reasoning stream authors against it.

The emission false floor (D1) and the phantom-need fault (D2) both turned on which KPIs a probe actually produces, so the battery was rebuilt and frozen. The v2 KPI battery draws call_required and honest_stop from both physics held-out and venue held-out families, balanced at 12 physics and 12 venue emission prompts and 6 physics and 6 venue honest-stop prompts. Every prompt is held out. The battery is frozen once, so a capability number moves only when the model moves.

The grounded library was materialized to 59 domains. *deep_full* carries 11,820 training rows and 6,000 evaluation rows at k_oracle 200, with zero leaks, and each row is stamped with its domain and its oracle. The venue corpus adds 13 families across roughly 7,316 documents and 18 independent tests, under the same leak law. Re-tagging the 13 venue families and *deep_full* with an oracle-field tagger is what let LOOM carve a per-tag held-out slice for the product half of the corpus, which D3 had silently disabled.

The last adaptation is persistence. The steering state now survives a restart: ATLAS family weights, the autopilot ema and *p_struct*, LOOM tags, family velocity, and the admission ledger are written under a single *steer* key and restored on resume. A run that stops at step 7133 and resumes does not relearn its mixture from a cold controller.

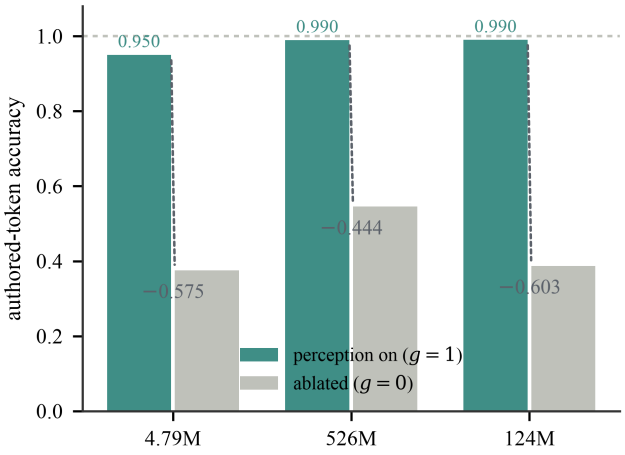

Figure 2. The founding ablation across the three runs in the lineage. Held-out authored-token accuracy with the perception stream on and ablated. The collapse stays large at every scale, between +0.44 and +0.60, and never falls back toward chance; the 124M value is measured after a knowledge injection. The 4.79M value is on 58 held-out rows; the 526M and 124M values are exact-architecture reloads.

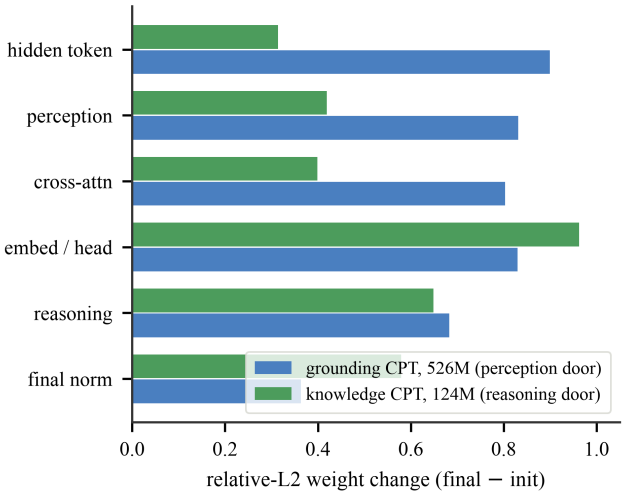

Figure 3. The two-door signature. Per-component relative weight change from two independent continued pretrains. The grounding pretrain moves the perception, cross-attention, and placeholder machinery most; the knowledge pretrain moves the embedding and reasoning path most. Each moved its own door most and the other's least.

6. The 1.939B run, reported as diagnostic-only

The 1.939B run is the largest in this lineage and the one that forces a correction. It launched from scratch on 2026-07-20, on the grounded spine plus the venue corpus lineage, and it was halted on 2026-07-21 at step 7133. We report it as diagnostic-only: it shows the architecture trains cleanly at this scale from scratch, and it shows nothing about capability. The reasons for reading it that narrowly are the rest of this section.

The configuration is *D2048, L28, H16, FFN8192*: 1.939B parameters, byte vocabulary 259, grouped-query attention with rotary embeddings and SwiGLU, trained from scratch. Venue holds 23.6% of the struct tier. Effective batch is 128 (batch 8 times gradient accumulation 16) at seq_len 2048, learning rate 1.8e-4,

warmup 3000 steps, cosine decay to 300k, and *p_struct* starting at 0.40. On one H100 the run held about 33.5k tokens per second and roughly 390 realized TFLOP/s. The last saved checkpoint is step 7133. Zero NaN across the run.

The clean signals

Under M2 meters the trajectory reads clean. Bits per byte fell from 3.517 to 1.114. GHLE authored-token accuracy climbed from 0.265 to 0.988. Language reached 0.956, coherence settled into the 0.72 to 0.82 band, and *know* moved across a 0.5 to 0.75 band without committing to a level. On the product side *foreign_tools* rose from 0.277 to 0.937 and *venue_brief* from 0.511 to 0.988. At the final canary every venue family scores between 0.988 and 0.9999, the physics families between 0.979 and 0.9997, and GHLE authored-accuracy holds at 0.988. These are evidence that the architecture trains at 1.939B from scratch and nothing more. No capability claim rides on them.

One number does not move. *toolcall* reads 0.00 at every checkpoint.

The model diagnosis: pre-liftoff

Here the teacher-forced and free-generation distinction has to be held exactly, because the two quantities live one column apart in the same canary record. Per-family accuracy is teacher-forced: given the true trace up to the CALL boundary, the model places high probability on the correct next bytes, and 0.99 there certifies that format and values are learned. *toolcall* is a different measurement. It is free-generation emission, from a cold prompt under greedy decode, asking whether the model emits a well-formed *>>>CALL* on its own. 0.00 means the run has not crossed the autonomous emission threshold. A generation peek confirmed no call was emitted. The same probe reads 1.0 on the 124M positive control, so the zero is honest-conservative, a real property of the 1.939B checkpoint at step 7133 rather than a broken meter. Prior runs lifted off emission in the tens of thousands of steps; this one stopped at 7133. The diagnosis is pre-liftoff rather than failure.

That diagnosis is about the model. What follows is about the experiment, and the two do not mix. The emission zero would still read the same on a perfectly instrumented run. The four limits below are the reason this run cannot be read as one.

Why the run is diagnostic-only

Contaminated curriculum signal. The autopilot *drive* at halt was 2.83, of which about 2.2 was phantom. Need was summed over three target KPIs, *decision*, *honest_stop*, and *echo*, that no probe on this run instrumented; each defaulted to *ema* 0.0 and was read as maximally starved. *p_struct* was pinned at the ceiling $p\_hi = 0.55$ to chase roughly 78% fictional starvation, with roughly 0.85 of the drive tracing to the genuinely starved measured *toolcall*. The steering acted on a signal that was mostly not real. This is D2 acting live in the archived telemetry. The fix, restricting need to the measured KPI set, is in code and unit-tested (P1 through P5 pass); it has not yet driven a run.

Inactive tag-level curriculum. LOOM requires a per-tag field to carve its eval slices. The venue families were registered without that field, so LOOM was silently inert across the product half of the corpus. Only the family-level lever, ATLAS, was live there. The run did not exercise the granular steering it was built to test.

Unfrozen mixture manifest. Streams and their weights were adjusted during the run without a versioned manifest. The telemetry therefore cannot be attributed to one reproducible curriculum, which is disqualifying for any capability claim even setting aside the first two limits.

Incomplete instrument set. WARP and WEFT, the consolidation-time and formation-time readouts, were built and self-tested but not wired into the loop. The run had no internal check on whether grounding was forming or being routed around, which is exactly the check the largest run most needed.

No external benchmark number is attached to the 1.939B run, and none is massaged into it. The figures above are internal canary measurements, read against a curriculum controller that was steering on a mostly phantom signal, with two of its four instruments dark. The run shows the architecture trains cleanly at this scale from scratch and nothing about capability. It is uncontrolled rather than failed, and the response it calls for is a controlled re-run.

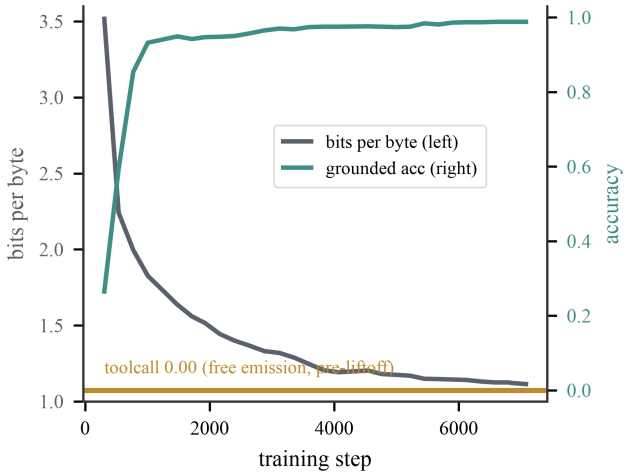

Figure 4. The 1.939B learning curve under the M2 meters. Bits-per-byte falls from 3.52 to 1.11 while grounded teacher-forced accuracy reaches ceiling by roughly step 3000. The free-emission tool-call probe is zero at every checkpoint.

Step	bpb	reason	lang	emit	foreign	brief	
309	3.517	0.265	0.694	0.00	0.277		0.511
1007	1.827	0.932	0.863	0.00	0.770		0.962
2162	1.444	0.948	0.955	0.00	0.790		0.966
3123	1.319	0.970	0.965	0.00	0.796		0.980
4067	1.192	0.975	0.962	0.00	0.816		0.985
5449	1.149	0.984	0.963	0.00	0.913		0.986
6584	1.125	0.988	0.951	0.00	0.929		0.988
7081	1.114	0.988	0.956	0.00	0.937		0.988

Table 3. Eight checkpoints from the 1.939B run under the M2 meters. Emission stays at zero while grounded accuracy and the discrimination measures climb.

7. What held, and the corrected protocol

Three results have held up under scrutiny, and none of them depend on the meter era, because each is read from checkpoint tensors or from a fixed held-out ablation rather than from a live probe.

The founding collapse persists at every scale and never falls back toward chance. At 4.79M the ablation of the perception stream at fixed weights took held-out authored-token accuracy from 0.9503 to 0.3756, a gap of 0.575. At the 526M CPT the same ablation read g=1 0.9898 against g=0 0.5455, a gap of +0.444, the smallest in the lineage and still most of the authored accuracy. At 124M the gap was +0.575, and after the knowledge-injection phase +0.603 (M2), with g=1 authored accuracy essentially unmoved at 0.9904. Injecting 29 GB of web, wiki, and Cosmopedia text did not let the reasoning stream route around perception. If anything the routing became marginally more causal.

The two continued-pretrains disagree about which parameters move, and they disagree cleanly. On the 526M grounding CPT the weight rel-L2 of final minus init ordered *hidden_token* 0.899 above *perception* 0.831, *tok/head* 0.830, *cross_attn* 0.803, *reasoning* 0.683, and *final_norm* 0.363. On the 124M phase-2 knowledge CPT the ordering inverts: *tok/head* 0.962 above *reasoning* 0.649, *final_norm* 0.579, *perception* 0.419, *cross_attn* 0.399, and *hidden_token* 0.314 at the bottom. A grounding objective and a knowledge objective leave mirror-image signatures on the same parameter blocks.

The 1.939B run is numerically clean. Zero NaN across the run to the last saved checkpoint at step 7133, at roughly 33.5k tok/s and about 390 realized TFLOP/s on one H100. The architecture is original and trained from scratch on a byte vocabulary of 259. That is the whole of what this run establishes. It shows the architecture trains cleanly at 1.939B from scratch, and it shows nothing about capability, because the experiment that was meant to test capability ran uncontrolled. Its autopilot steered on a *drive* of 2.83 that was roughly 78% phantom. Its tag-level controller, LOOM, was inert across the venue half of the corpus. Its mixture was changed in flight with no frozen manifest, so the telemetry cannot be pinned to one reproducible curriculum. And its two internal grounding readouts, WARP and WEFT, were never wired into the loop, so the largest run in the lineage had no live check on whether grounding was forming or being routed around. Section 6 states those four limits as facts about the run. The run is diagnostic-only: an uncontrolled experiment rather than a failed one, and the response to an uncontrolled experiment is a controlled re-run.

Now the open items, without cushioning.

Everything here is one run at one seed. There is no subword baseline to separate the contribution of the byte-level substrate from the contribution of the dual-stream objective, and until one exists the attribution stays partly open. The 1.939B run has not lifted off autonomous emission: *toolcall* reads 0.00 at every checkpoint, which Section 6 diagnoses as pre-liftoff, resting on the teacher-forced-versus-free-generation split rather than on failure; but pre-liftoff is still zero, and the run stopped at 7133 where prior runs crossed emission only in the tens of thousands of steps. The D2 phantom-need fix is written and unit-tested (P1 through P5 pass); it has not driven a run, so every archived 1.939B telemetry record, including the *drive* of 2.83 and *p_struct* pinned at 0.55, comes from the pre-fix controller. WARP and WEFT are built, self-tested, and in WEFT's case run against the 524M as a fit-and-ratify probe on declination_deg; neither is yet wired as a live in-loop callback.

That is the state.

The corrected protocol is set by the four integrity limits, and it is procedural rather than clever. Freeze one configuration and one versioned mixture manifest, both written to disk before launch and neither edited after. The 1.939B run had the config but not the manifest, which is exactly why its telemetry cannot be attributed to a single reproducible curriculum. Diff that config against the 124M once, and read it as a diff. Dry-run the worst-case branch once, which for this architecture is the gate-1 struct forward at full context, the branch D4 proved will OOM if you preflight the cheap path instead. Before launch, verify every instrument is live instead of assuming it: the *measured* KPI set complete so the *drive* carries no phantom need, an oracle/tag field present on every family so LOOM is not silently inert on half the corpus, and the WARP and WEFT callbacks active so the largest run finally has an internal read on whether grounding is forming or being routed around. Then one launch, with no live mutation of streams, feed order, or weights while the run is in flight; the autopilot ema, *p_struct*, ATLAS weights, LOOM tags, family velocity, and the admission ledger persist under the *steer* key and are read at resume, not rewritten by hand mid-run.

Across this lineage the recurring failure came from changing a run while it ran and losing the record of what changed, not from the architecture and not from any single bug. A frozen config and a versioned manifest are unglamorous, and they are the discipline the next run has to hold.

Sources. All artifacts are local and hash-verified where checkpoints are referenced: the scale-lineage record, the full 1.939B telemetry archive, the controller and probe modules, the product simulator and corpus generators, the fifty-nine-domain grounded library, the orchestrator specification, and the interpretability design specs. The baseline is the 4.79M ablation reported in Grounded Dual-Stream Language Models, preprint, July 2026. Point values, single seeds, no state-of-the-art claims.