

The Name Is the Bridge: Latent Injection into Frozen LLMs Fails in Principle, Reflection Lives in the Argmax, and a Rented Prior Only Pays In-Language

Cercle Digital (independent research)
Correspondence: research@cercle.digital

Abstract

Can an agent that thinks in a continuous latent space use a *frozen* market language model (an LLM whose weights are never updated) as a reflection organ, without ever decoding tokens? We answer with a chain of sharp pre-registered negatives and the positive interface they force, measured on a frozen 1.5B-parameter instruction model against honest zero-LLM baselines. **(1)** Latent-to-latent reflection (agent latent, projected into the LLM’s embedding space, one frozen forward, a learned linear read of the last hidden state back to the latent) adds nothing over the frozen encoder’s own vector geometry (0.219 ± 0.050 vs. zero-LLM arithmetic 0.641 on relational analogies), while letting the same model emit one word (token chain-of-thought) reaches 0.859 and a weight-shuffled control collapses to chance: the reasoning prior exists, but a linear read of a single hidden state discards it. **(2)** Every continuous zero-token extraction fails: the expected embedding under the model’s own next-token distribution (the answer *without* the argmax, the discrete highest-probability choice) is near chance (0.038), and latent autoregression *degrades* with depth ($0.192 \rightarrow 0.112$): the answer lives in the argmax, the discrete selection, not in any continuous average of the model’s state. **(3)** Multi-hop depth is carried by *discretising through the agent’s concept inventory between hops* (0.900 vs. 0.100 for a continuous intermediate), pays accuracy at three hops (1.000 vs. 0.750 flat), and buys per-hop verification: an injected hallucination propagates unchecked without grounding (0.000) and is fully repaired by it (1.000). **(4)** Injection itself resolves negatively in principle: a *nameless* latent is not injectable (held-out bridges sit at chance, 0.031, even for a functionally trained non-linear projector, 0.078 ± 0.064), because the model’s manifold is organised by token identity, not meaning; yet a *named* concept injected as its own tokens recovers $0.812 \approx$ chain-of-thought. Hence the canonical interface: perceive, admit, *name*, inject the name’s tokens, let the frozen model score the inventory, select discretely, verify each hop. **(5)** Deployed as a swappable proposer under execution-owned verification, the rented prior pays only where choices are predictable from language (a program-search world: 16/16 solved at 3.7 verifications vs. 3/16 at 13.0 random), is inert where only interaction knows the affordances (a real text adventure: 18 vs. 17 rooms, identical rejections), is matched by a $40\times$ smaller owned encoder (2.1 vs. 3.1 verifications), and its self-confidence is not a reliability signal (confidently wrong at $p_{\text{top}} = 0.971$; the accuracy of confidence-gated commits is exactly the prior’s, 1.000 aligned, 0.000 contradicted). A self-contained code supplement reproduces every number.

1 Introduction

A growing line of work makes agents *think in latent space*: plan, compare and compose in a continuous embedding, and touch tokens only at the periphery (Hao et al., 2024; Deng et al., 2023; Goyal et al., 2024). A separate, older economy makes agents *rent* capability from a large pretrained language model. This paper measures the seam between the two: if an agent already owns a grounded continuous concept space, can it plug a frozen market LLM in as a pure *reflection organ*, injecting thoughts as vectors and reading thoughts back as vectors, with zero tokens decoded? The question matters because every token avoided is latency, supervision surface, and architectural purity saved; and because if the answer is no, the failure mode should be stated precisely enough to fix the interface once and for all.

The lock, stated as one research question: *where, along the path from a continuous concept to a frozen LLM’s rea-*

soning prior and back, does the information actually live: in the vectors, in the hidden states, in the output distribution, or in the discrete token selection? Existing work does not answer it. Soft-prompt and steering methods (Lester et al., 2021; Li & Liang, 2021; Subramani et al., 2022; Turner et al., 2023; Zou et al., 2023) optimise continuous inputs *against a task loss on the same model*, so they never test whether an *external* semantic vector is injectable without task training. Hidden-state lenses (nostalgebraist, 2020; Belrose et al., 2023; Ghandeharioun et al., 2024) read what a hidden state contains, but not whether a single linear read can replace the autoregressive process. And latent-reasoning work (Hao et al., 2024) loops the model’s *own* hidden states after end-to-end training, which leaves the frozen, foreign-latent regime unmeasured.

Terminology, up front. Five terms recur from the title onward. A *frozen* LLM is used forward-only; its weights are

never updated (the same discipline one applies to a frozen vision or sentence encoder). The agent’s *latent* is the 384-dimensional embedding of a frozen sentence encoder (Wang et al., 2020; Reimers & Gurevych, 2019); it stands in for the continuous concept space of a latent-thinking agent, and the agent’s *concept inventory* is the finite set of concepts it has admitted, each carrying a *name* (a word). The *argmax* is the discrete highest-probability selection from the model’s output distribution, what autoregressive decoding commits to at each step; *token chain-of-thought* (CoT) lets the model emit tokens as its medium of thought (Wei et al., 2022; Kojima et al., 2022). Finally, *verification* (also *touch*, written `do()` when it acts on a world) is a check the agent owns outside the LLM: probing a game state on a copy, executing a program on examples, or consulting a stored relation (Pearl, 2009); following Belnap’s four-valued logic (Belnap, 1977), a claim contradicted by verification is retained as *Both* (asserted and refuted, kept as a flagged contradiction) rather than silently averaged away.

We proceed as a chain of pre-registered measurements, each closing one door and opening the next, on one deliberately stripped substrate (relational analogies $A:B :: C:?$, then multi-hop relational chains), with honest baselines: the zero-LLM vector arithmetic of Mikolov et al. (2013), an oracle-token arm that isolates injection from extraction, a weight-shuffled model that isolates the trained prior from architecture and compute, and token CoT as the medium we are trying to avoid. Concretely:

1. **Latent-to-latent reflection adds nothing** (Section 4). The bridge reaches 0.219 ± 0.050 , below zero-LLM arithmetic (0.641) and far below token CoT (0.859); the shuffled control at $0.035 \pm 0.024 \approx$ chance proves the prior is real and the reads lose it. The loss decomposes: injection +0.196, extraction medium +0.444 (dominant).
2. **The answer lives in the argmax** (Section 5). The expected embedding under the model’s own output distribution scores $0.038 \approx$ chance even with oracle inputs; latent autoregression degrades with depth ($0.192 \rightarrow 0.112$). Scoring the agent’s own concept inventory (a sharp selection, zero tokens emitted) recovers 0.500 with clean inputs: extraction is solved by selection, and the wall migrates to injection.
3. **Depth passes through the discrete collapse** (Section 6). A two-hop chain with a continuous intermediate collapses (0.100); discretising the intermediate through the inventory reaches 0.900. At three hops the decomposition pays accuracy (1.000 vs. 0.750 flat) and per-hop verification owns the correction under an injected hallucination ($0.000 \rightarrow 1.000$).
4. **The name is the bridge** (Section 7). A named concept injected as its own tokens reaches $0.812 \approx$ CoT; the injection wall decomposes into the mono-vector collapse (-0.312) and linear shrinkage (-0.219). A *name-*

less latent is not injectable: held-out bridges sit at chance (0.031), and the strongest continuous arm we could build (a non-linear projector functionally distilled against the model’s own behaviour, converged) stays at chance (0.078 ± 0.064). The dilution is intrinsic, not a tuning failure.

5. **What the rented prior buys, plugged in correctly** (Section 8). As a swappable proposer under execution-owned verification it pays only *in-language* (where the choice is predictable from text): $3.5\times$ fewer verifications and $5\times$ the coverage on program search, inert on a real text adventure. A $40\times$ smaller owned encoder matches it, and its confidence is not a safety signal.

Scope claim, stated up front. We claim no state-of-the-art number and no new learning mechanism. The contribution is a *measured demarcation*: which channels between a foreign continuous concept space and a frozen LLM carry information (name tokens in, discrete selection out) and which do not, even in principle (raw latents in, continuous averages out); plus the interface this demarcation forces and a deployment-grain audit of what the rented prior is then worth. All results are on one frozen 1.5B model (Qwen Team, 2024) and relational tasks; the declared bounds are in Section 10.

2 Related work

Latent and continuous reasoning. Coconut (Hao et al., 2024) feeds the model’s last hidden state back as the next input embedding and shows latent chains can beat token CoT after *end-to-end training* of the loop. Our negative results do not contradict this; they bound the neighbouring regime Coconut does not measure: a *frozen* model and a *foreign* latent (an external encoder’s concept vector) for *held-out* concepts. There the continuous channel carries nothing, even with a trained non-linear projector (Section 7). Implicit CoT distills reasoning into hidden states (Deng et al., 2023); pause tokens buy computation without content (Goyal et al., 2024). Both train the model itself; we hold it frozen by design, because the agent’s economy is to rent the prior, not to own its training.

Soft prompts, steering, representation engineering. Prompt tuning (Lester et al., 2021) and prefix tuning (Li & Liang, 2021) prove continuous inputs can *control* a frozen model when optimised against a task loss; steering vectors (Subramani et al., 2022; Turner et al., 2023) and representation engineering (Zou et al., 2023) add directions extracted from the model’s own activations. All operate *inside* the model’s manifold. Our injection question is different: can a vector from *another* model’s semantic space designate a concept the bridge never saw? The held-out answer is no

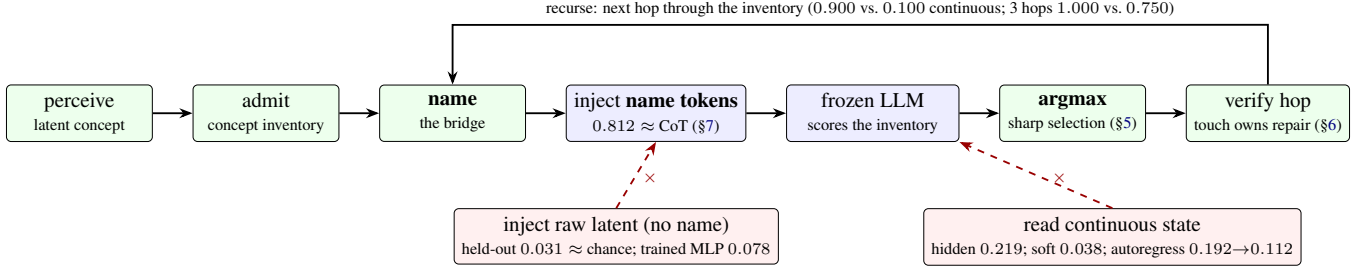


Figure 1: The canonical interface, and the measured dead ends that force it. Top lane: a concept becomes reflectable once *named*; its name’s tokens are injected, the frozen LLM scores the agent’s own concept inventory, the discrete argmax selects, and the agent’s own verification (touch) checks every hop, repairing injected hallucinations (0.000 $\rightarrow$ 1.000, Section 6). Bottom lane, dashed: the two channels that fail, injection of a nameless latent (at chance held-out, even functionally trained; Section 7) and every continuous read of the model’s state (Section 5). Numbers are hit@1 on the relational-analogy bench except where noted; chance = 0.028.

(chance), which we read with Elhage et al. (2022) and Geva et al. (2021): the input manifold is organised by token identity (spelling), not by meaning, so no generalising map from meaning to embedding exists to be learned.

Reading hidden states. The logit lens (nostalgebraist, 2020), tuned lens (Belrose et al., 2023) and Patchscopes (Ghandeharioun et al., 2024) decode intermediate states, and Patchscopes in particular shows a hidden state *patched into a decoding context* can be verbalised. Consistent with that, our results show the information is recoverable by *selection* (scoring candidates through the frozen forward) and lost by *linear averaging* (a single-hidden read, or the expected embedding): the medium that unlocks the prior is the discrete choice, which is also what the expressivity account of CoT predicts (Merrill & Sabharwal, 2024).

Agents, tools, verification. ReAct (Yao et al., 2023) and Toolformer (Schick et al., 2023) put the LLM in the controller seat, calling tools. Our deployment sections invert the geometry: the LLM is an *organ the agent scores*, proposing candidates under a verifier the world owns (execution, probing), in the spirit of pairing language models with structured knowledge (Pan et al., 2024). The calibration contrast is direct: models can self-assess in-distribution (Kadavath et al., 2022), but on lived facts that contradict the prior we measure confident wrongness ($p_{\text{top}} = 0.971$ on a false claim), so commit rights cannot be delegated to confidence (Section 8).

3 Setup and metrology

The two frozen models. The *agent side* is a frozen 22M-parameter sentence encoder (all-MiniLM-L6-v2; Wang et al., 2020; Reimers & Gurevych, 2019) whose 384-d embedding is the agent’s latent concept space. The *rented side* is Qwen2.5-1.5B-Instruct (Qwen Team, 2024), held exactly like a frozen encoder: forward-only, never fine-tuned, no gradient ever enters it. Every learned component in this

paper is a cheap adapter *outside* both models (closed-form ridge regressions, one small MLP), so that what is measured is the interface, never the models’ training.

The analogy bench. Relational analogies $A:B :: C:?$ over a hand-built offline table: five world-knowledge relations a rented prior should carry (capital, currency, colour, animal sound, tool use) and three morphological relations the encoder’s geometry already carries (plural, opposite, comparative); 8 quads per relation, 64 analogies, an answer pool of 36 concepts, chance = $1/36 = 0.028$; the three operand words are always excluded from the pool. This is deliberately the most stripped reflection task that still *requires* a world prior, and its zero-LLM baseline is strong (0.641), which is the honest comparator a rented organ must beat.

Arms shared across benches. *Zero-LLM arithmetic:* the parameter-free $b - a + c$ retrieval of Mikolov et al. (2013) in the agent latent. *Zero-LLM ridge:* a learned readout from the concatenated operand latents (same learning machinery as the bridge, no LLM). *Oracle tokens in:* the operands enter as their real tokens (isolates extraction from injection); zero tokens are ever emitted. *Shuffled control:* every weight tensor of the LLM is randomly permuted (destroys the trained prior, preserves architecture and compute). *Token CoT:* the model generates one word, re-embedded and retrieved (the medium the design rejects, kept as the contrast arm).

3.1 Metrology

All randomness is seeded and explicit; quads and worlds regenerate bit-for-bit from (configuration, seed). Arms with a learned readout are evaluated on random 60/40 train/test splits over 10 seeds and reported as mean $\pm$ 95% CI (1.96 SE); deterministic arms (teacher-forced scoring at temperature 0) are reported with bootstrap 95% CIs over items, and we flag the two small- n benches (10 and 8 queries) where those intervals are wide. Baselines are strong by construction (the zero-LLM comparator is never a strawman),

controls are adversarial (oracle inputs, shuffled weights), and negative results are pre-registered and kept as results. Each bench name below is the name of the script that reproduces it in the code supplement (Appendix A).

4 Negative 1: latent-to-latent reflection adds nothing

Hypothesis (pre-registered). A frozen LLM can serve as a latent-to-latent reflection operator: project the agent’s operand latents into the LLM’s embedding space with a cheap bridge U (ridge, closed-form), run one frozen forward over the analogy scaffold, read the last hidden state back to the agent latent with a learned readout V ; zero tokens decoded. If the rented prior is reachable this way, the bridge should beat the encoder’s own geometry (zero-LLM arithmetic) and approach token CoT.

Experiment (operationalisation). Bench LATENT-BRIDGE. U : ridge from the 384-d latent to the LLM’s native word embedding (mean over sub-tokens), fitted on the bench vocabulary. Scaffold in embedding space: $[U(A) | \text{“is to”} | U(B) | \text{“as”} | U(C) | \text{“is to”}]$. V : PCA + ridge from the last hidden state to the answer latent, refit per seed on the train split; answer retrieved by nearest neighbour in the agent latent. Controls as in Section 3.

Result. Table 1. The bridge reaches 0.219 ± 0.050 : below the zero-LLM arithmetic 0.641 and far below token CoT 0.859. The prior itself is real and substantial: CoT $\gg$ arithmetic, and the shuffled control sits at chance (0.035 ± 0.024), so the value is the trained weights, not the transformer’s architecture or compute. The loss decomposes cleanly: injection (oracle – bridge) = +0.196; extraction medium (CoT – oracle) = +0.444, the dominant term. Per relation, the bridge peaks weakly on colour (0.38) and capital (0.39) and collapses to 0.0 on animal sounds, while arithmetic stays unbeaten on morphology (plural/comparative 1.0): the encoder already carries what is morphological, and the bridge recovers only scraps of what is world knowledge.

Conclusion. Pre-registered null, kept: a single frozen forward read linearly from one hidden state does not deliver the reasoning prior, and the dominant loss is the *read*, not the bridge in. The prior unlocks through the autoregressive process, not through a linear read of its state. The next section asks what, in that process, carries the information.

Table 1: Bench LATENT-BRIDGE: held-out hit@1, 64 analogies, 10 seeds, chance 0.028. The reflection prior exists (CoT vs. shuffled control) and the latent bridge does not reach it.

arm	hit@1
zero-LLM arithmetic ($b - a + c$)	0.641
zero-LLM ridge readout	0.177 ± 0.041
latent bridge (frozen forward, 0 tokens)	0.219 ± 0.050
oracle tokens in (0 tokens out)	0.415 ± 0.054
shuffled-weights control	0.035 ± 0.024
token CoT (contrast arm)	0.859

5 Negative 2: the answer lives in the argmax

Token CoT differs from the latent bridge by exactly two operations: it *selects* (argmax over the vocabulary) and it *re-encodes* the selected token. Both can be imitated continuously; if either continuous imitation matches CoT, tokens were never needed.

Hypothesis (pre-registered). (a) The *soft-distribution read*, the expected input embedding under the model’s own next-token distribution ($e = p^\top W_{\text{emb}}$, exactly CoT minus the argmax), recovers what CoT has. (b) *Latent autoregression*, re-injecting that soft token as the next input embedding for K steps and reading the final hidden state, composes latent reflection with depth.

Experiment (operationalisation). Bench ZERO-TOKEN-READS: both reads, $K \in \{1, 2, 4\}$, injected and oracle-token inputs, learned readout per seed as before. Bench INVENTORY-SCORING: the architecturally matched alternative, in which the frozen model *scores the agent’s own concept inventory* (teacher-forced, length-normalised log-probability of each candidate concept after the analogy prefix; one forward per candidate, argmax over concepts, zero tokens emitted).

Result. Figure 2. Every read that refuses to choose fails. The soft-distribution read is at chance, 0.038, *identically with oracle inputs*: the expected embedding is a quasi-invariant blur. Latent autoregression starts at 0.192 ($K=1$) and *degrades* to 0.112 ($K=2$) and 0.135 ($K=4$): the blur composes. The single-forward hidden read, recomputed in this harness, stays at 0.277. By contrast the sharp selection over the inventory reaches 0.500 with oracle inputs (injected: $0.281 \approx$ the hidden read), against CoT 0.859.

Conclusion. The reflection answer lives in the *argmax*: the discrete selection, not any continuous average of the model’s state. This settles extraction (score candidates, select sharply) and migrates the remaining wall to injection:

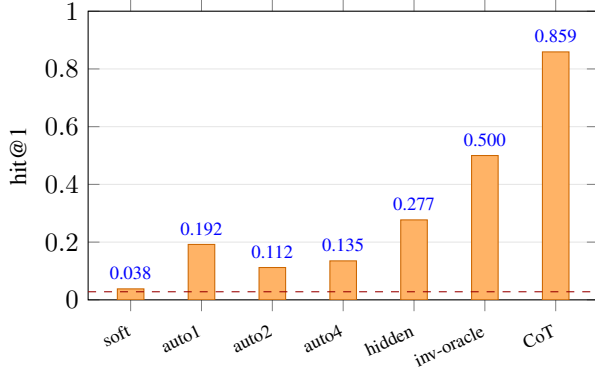


Figure 2: Where the answer lives (benches ZERO-TOKEN-READS and INVENTORY-SCORING; 64 analogies; dashed line: chance 0.028). Continuous reads of the frozen model’s state (the soft-distribution expected embedding; latent autoregression at $K=1, 2, 4$; the single-hidden linear read) all sit far below the two discrete arms: sharp selection over the concept inventory with clean inputs (0.500) and full token CoT (0.859). Autoregression *degrades* with depth: the blur composes.

with clean inputs the inventory scoring reaches 0.500, with the ridge-injected operands only 0.281 (gap +0.219). It also explains, in advance, why flat continuous depth cannot work: composing reads that each dilute must degrade, which Section 6 confirms in positive form.

6 Positive 1: depth passes through the discrete collapse

If the answer lives in the argmax, then multi-hop reflection should be built by *recurring through the agent’s inventory*: select a concept discretely at each hop and feed the *name* of the selection to the next hop, instead of chaining continuous states inside the model.

Hypothesis (pre-registered). (a) On a two-hop task, a gated chain (discretise the intermediate through the inventory) beats the same chain with a continuous intermediate. (b) On a three-hop task, the gated decomposition pays *accuracy* over a flat single read. (c) Per-hop verification owns the correction: an injected wrong hop is repaired when each selected edge is checked against the agent’s stored relations, and propagates when it is not.

Experiment (operationalisation). Bench GATED-DEPTH: currency $\circ$ capital⁻¹ over a 10-country table (“the currency of the country whose capital is X ”); arms *flat* (one read), *gated* (select the country from the inventory, then its currency), *no-gate* (the intermediate slot filled with the continuous hidden state of hop 1). Named concepts enter as their own tokens (oracle input) to isolate depth from

injection. Audit arms: gate margin on out-of-inventory queries; two-channel Belnap bookkeeping. Bench DEPTH3-VERIFICATION: monument $\rightarrow$ city $\rightarrow$ country $\rightarrow$ currency over an 8-row chain; stress arm forces a wrong concept at hop 2 (an injected hallucination) with verification ON vs. OFF; the verifier is the stored relation table (exact here, a declared idealisation).

Result. Figure 3. (a) Gated 0.900 vs. continuous intermediate 0.100 ($\Delta = +0.800$): the discretising collapse of the gate is what carries depth; refusing it (staying continuous) destroys the signal, the positive image of the autoregression decay of Section 5. At two hops accuracy is null against flat (0.900 vs. 0.900): the model one-shots that depth, and the honest value of the decomposition is per-hop verifiability. (b) At three hops depth pays: gated 1.000 vs. flat 0.750 (the flat read misses two of eight chains; the decomposition recovers both). (c) Under the injected hallucination at hop 2: verification OFF 0.000 (all eight corrupted chains commit a wrong final answer), verification ON 1.000 (the invalid edge is caught, retained as Both, and the touch-consistent candidate is re-selected), $\Delta = +1.000$. Audit: the gate margin separates in-inventory from out-of-inventory queries by +8.68 on average (a calibratable abstention signal); the zero-LLM chaining floor is 0.70 (the rented organ adds chaining, and remains removable); agreement files as True, injected disagreement as Both (the two channels never collapse into one scalar).

Conclusion. Depth belongs to the *interface*, not to the model’s continuous grain: recurse through the inventory, verify each hop. The correction is owned by the verifier, never by the LLM: an hallucinated hop is gated and repaired, not committed. Declared bounds: $n=8$ at three hops (bootstrap CI on the flat arm $\approx \pm 0.30$), and the verifier here is an exact relation table; in a live agent the verifier is a lived, noisy graph.

7 Negative 3 and Positive 2: the name is the bridge

One wall remains: injection. Sections 4 and 5 measured it as the oracle-vs-injected gap (+0.196 to +0.219). We now decompose it, close it for named concepts, and show it cannot be closed for nameless ones.

Hypothesis (pre-registered). (a) The injection loss splits into the *mono-vector collapse* (averaging a word’s sub-token embeddings into one vector) and the *linear shrinkage* of the ridge bridge. (b) If the bridge is denied the concept’s name (trained on an auxiliary vocabulary, evaluated on held-out task concepts: the “nameless percept” simulation), a multi-token regression to real sub-token targets still recovers in-

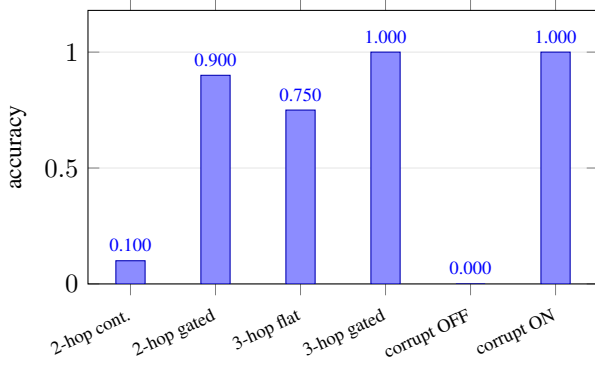


Figure 3: Depth passes through the gate, and verification owns the repair (benches GATED-DEPTH, 10 queries, and DEPTH3-VERIFICATION, 8 queries; bootstrap CIs in text). A continuous intermediate collapses (0.100) where the discretised one chains (0.900); three-hop decomposition pays accuracy over the flat read (1.000 vs. 0.750); an hallucination injected at hop 2 propagates without verification (0.000) and is fully repaired with it (1.000).

jection. (c) If (b) fails, the strongest continuous arm, a non-linear projector *functionally distilled* against the frozen model’s own behaviour, recovers it.

Experiment (operationalisation). Bench INJECTION-DECOMPOSITION: inventory-scoring evaluation of (i) oracle multi-token (the name’s real sub-token sequence), (ii) oracle mono-vector (the same embeddings averaged), (iii) the linear mono-vector bridge (in-sample). Bench NAMELESS-INJECTION: the linear bridges retrained on 188 auxiliary words with all 88 task concepts held out, single- and multi-token ($K=3$) targets. Bench LEARNED-PROJECTOR: an MLP ($384 \rightarrow 512 \rightarrow K \times H$, $K=2$) trained to minimise the KL divergence between the frozen model’s next-token distribution on the injected soft tokens and on the word’s real tokens (three probe templates, gradient clipped, LLM frozen), same held-out split. An earlier unclipped run of this distillation diverged; with clipping it converges (KL $0.663 \rightarrow 0.130$ over 15 epochs), so the verdict below is not a training failure.

Result. Figure 4. (a) A named concept injected as its own tokens reaches $0.812 \approx \text{CoT } 0.859$: *for named concepts there is no injection problem*. The wall decomposes as mono-vector collapse $0.812 \rightarrow 0.500$ (-0.312) plus linear shrinkage $0.500 \rightarrow 0.281$ (-0.219); the earlier 0.281 was also flattered, in hindsight, by the bridge having seen the task words (in-sample memorisation). (b) Held out, both linear bridges fall to chance: single 0.031, multi-token 0.031, against the oracle’s 0.812. (c) The functionally distilled MLP, evaluated held out, stays at chance: 0.078 ± 0.064 (the interval includes chance 0.028), despite converged training.

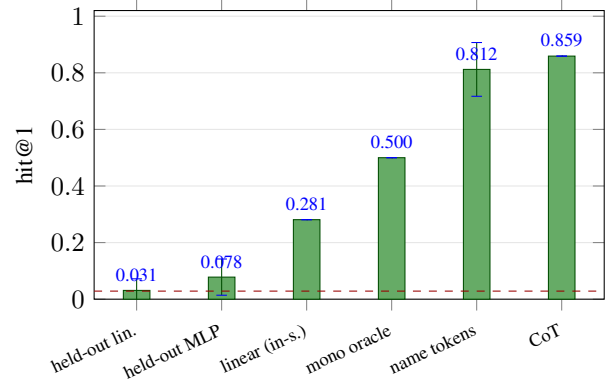


Figure 4: The injection ladder: the name is the bridge (benches INJECTION-DECOMPOSITION, NAMELESS-INJECTION, LEARNED-PROJECTOR; dashed line: chance 0.028). Held out (the nameless case), both the linear bridge and the functionally distilled MLP projector sit at chance; the in-sample linear bridge only memorised. With the name, the mono-vector average of its tokens reaches 0.500, and the full name-token sequence reaches $0.812 \approx$ token CoT: injection is solved by naming, and only by naming.

Conclusion. The dilution is *intrinsic* to the continuous channel, not a tuning failure: from the semantic latent alone, nothing exploitable reaches the model for a concept the bridge never saw. The reading is information-theoretic rather than optimisational: the agent latent encodes *meaning*; the frozen model’s input manifold is organised by *token identity* (Elhage et al., 2022; Geva et al., 2021); and one cannot predict a never-seen word’s spelling from its meaning. Architecturally this closes the loop: a percept becomes reflectable once *named*. The naming faculty is not a step one may shortcut by injecting raw latents; it is the bridge itself. The canonical interface of Figure 1 follows: perceive $\rightarrow$ admit $\rightarrow$ name $\rightarrow$ inject the name’s tokens $\rightarrow$ frozen reflection $\rightarrow$ score the inventory $\rightarrow$ discrete selection $\rightarrow$ verify per hop.

8 The rented prior at deployment grain

The interface being fixed, what is the rented organ worth once plugged in correctly? We measure three deployment questions on worlds where the *verifier is owned by the world*: a propose-verify search in which a swappable proposer orders candidate options and the world’s own check (a move probed on a copy of a real game; a program executed on shown examples) accepts or rejects each one. A rejected option is never committed, so correctness is owned by verification on every arm; the proposer can only buy *efficiency* (fewer verifications) and *coverage*.

8.1 Where the prior pays: in-language domains

Hypothesis (pre-registered). The rented proposer pays exactly where the choice is predictable from language (the option’s text matches the goal’s text), and is inert where validity is not recoverable from the observed text (observation under-determines the affordances).

Experiment (operationalisation). Bench PROPOSER-DOMAINS: one search, two worlds, proposer swapped between the frozen LLM (PMI-normalised continuation scoring of each option against goal + state description; the option’s null-prompt score is subtracted, removing token-frequency bias) and random. World 1: the 1977 Colossal Cave text adventure (verifier: a movement probed on a deep copy; the room prose does not say which exits are real). World 2: natural-language→program search over 144 two-step list-manipulation programs whose specs name the operation family but drop parameters (verifier: execution on shown examples).

Result. Table 3, top. Programs: the LLM proposer solves 16/16 goals at 3.7 verifications-to-solve; random solves 3/16 at 13.0 ($3.5\times$ fewer verifications, $5\times$ the coverage). Adventure: 18 rooms (LLM) vs. 17 (random) at the same verification budget, with *identical* rejection counts (73 vs. 73): the prior orders the candidate directions no better than chance, because exit validity is simply not in the prose. Zero garbage is committed on either side of either world: every hallucinated option dies at the verifier.

Conclusion. The rented prior accelerates the general propose-verify faculty only *in-language*, where its knowledge predicts the choice; where only interaction knows (under-determined affordances), it is inert, and the verifier carries the correction in both regimes.

8.2 Rented vs. owned, at equal budget

Hypothesis (pre-registered). On the one terrain where the rented prior pays (program search), a proposer built from the agent’s own frozen 22M sentence encoder (cosine between the spec’s embedding and each program phrase’s embedding) matches the 1.5B rented model.

Result. Bench RENTED-VS-OWNED, Table 3 bottom: over 24 goals at budget 40, random solves 11/24 at 23.0 verifications; the rented LLM 24/24 at 3.1; the owned encoder 24/24 at 2.1, matching and slightly beating the rented model while being roughly $40\times$ smaller in resident size (about 80 MB vs. 3 GB). Zero garbage on all three arms.

Conclusion. Even as a pure efficiency lever, the large rented prior adds nothing decisive over the small encoder already owned: the ordering work a proposer does here is done as well by the agent’s own frozen embedding. Declared bound: this world’s specs are close to string matching between spec and program phrase, which favours a sentence encoder; a task whose candidate ordering requires deep reasoning could reopen the gap, but it is precisely the LLM-favourable case (program search) that falls.

8.3 Confidence is not a commit signal

Hypothesis (pre-registered). If the model’s self-confidence tracks reliability, a commit-rights dial h (an unverified claim may be committed, labelled as a hypothesis, iff $p_{\text{top}} \geq 1 - h$, where p_{top} is the softmax confidence over the candidate set) should trade coverage against accuracy smoothly, and a strict threshold should filter out the claims that contradict the agent’s lived world.

Experiment (operationalisation). Bench CONFIDENCE-GATE: twelve country→capital items the agent never verified, half *aligned* with the prior, half *contradicted* by construction (in this world the lived edge for France points to Tokyo: the agent’s world model disagrees with the pre-trained prior, the situation the dial exists for); three verified items as the touch control; sweep $h \in \{0, 0.2, \dots, 1\}$.

Result. Table 2. At $h=0$ nothing unverified commits (100% abstention). By $h=0.2$ all twelve items commit, and the dose-response is flat from there on: committed accuracy 0.500 overall, split as 1.000 on aligned items and 0.000 on contradicted ones. The mechanism is measured directly: the model is ultra-confident even when wrong for this world ($p_{\text{top}} = 0.971$ for the claim contradicting the lived France edge; the contradicted half spans 0.867–0.999), so no confidence threshold separates the false half. Verified items commit correctly at every h .

Conclusion. Self-confidence is not a reliability signal at this grain: the guard $p_{\text{top}} \geq 1 - h$ does not protect, which sharpens the in-distribution self-knowledge picture of Kadavath et al. (2022) at exactly the point an agent needs it (claims its lived world contradicts). Commit rights therefore stay with verification; $h=0$ is the safe default, and any committed hypothesis stays labelled, never a fact.

9 Discussion

What kind of negatives are these? Three of the five results are nulls, and they are the load-bearing ones, because each is a null *with its mechanism measured*. The latent

Table 2: Bench CONFIDENCE-GATE: dose-response of confidence-gated commits on unverified items (6 prior-aligned, 6 prior-contradicted). The response is all-or-nothing and flat: confidence never separates the false half.

h	committed	abstained	acc. aligned	acc. contra
0.0	0.00	1.00	–	–
0.2	1.00	0.00	1.000	0.000
0.4–1.0	1.00	0.00	1.000	0.000

bridge does not fail for lack of a prior (the shuffled control isolates a real prior) but because a linear read of one hidden state discards what the autoregressive selection preserves. The continuous reads do not fail for lack of training (the functionally distilled projector converges and still sits at chance held-out) but because meaning does not determine spelling: no map from a foreign semantic space into a token-identity-organised manifold can generalise to unseen names. That is a statement about the *geometry of the interface*, not about our adapters.

Relation to Coconut and latent reasoning. Our nulls bound, rather than contradict, continuous-thought training (Hao et al., 2024): Coconut loops the model’s *own* hidden states (same manifold, in-distribution) and trains the loop end-to-end; we inject a *foreign* latent into a *frozen* model and evaluate held-out concepts. Both can be true at once, and together they draw a clean boundary: the continuous channel works *inside* a model that is trained for it; the bridge *from an external grounded concept space into a frozen model* requires the name. For agents that rent rather than train their LLM, the second regime is the one that matters.

The LLM as a scored organ, not a controller. The interface that survives measurement inverts the usual tool-use geometry (Yao et al., 2023; Schick et al., 2023). The model never acts, plans, or commits; it *scores the agent’s inventory* (a sharp selection the agent owns) and *proposes orderings* under a verifier the world owns. Everything safety-relevant stays outside it: depth (recursion through the gate), correction (per-hop verification; 0.000 $\rightarrow$ 1.000 under injected hallucination), and commit rights (never delegated to confidence; Section 8.3). What the rented organ contributes is measurable and bounded: chaining it does not own (+0.20 over the zero-LLM floor at two hops), accuracy at depth (+0.25 at three hops), and in-language ordering efficiency (3.5 $\times$), the last of which a 40 $\times$ smaller owned encoder matches.

Naming as an architectural primitive. The practical corollary of Section 7 is that a naming faculty (assigning a stable token string to an admitted concept) is not a convenience for human interaction; it is the *only* generalising chan-

nel into a frozen LLM’s prior. Grounding, admitting, and naming are therefore not preprocessing that a sufficiently clever projector will one day shortcut; on the evidence here, they are the bridge.

10 Limitations

- **One frozen LLM.** Every measurement uses Qwen2.5-1.5B-Instruct. The mechanisms argued (argmax selection, token-identity organisation of the input manifold) are not model-specific claims, but their generality is unproven here: replication on a second frozen model is required before any cross-model claim, and is the first item of future work.
- **Relational tasks, small tables.** The analogy bench (64 quads, pool 36) and the hop tables (10 and 8 rows) are deliberately stripped; the two depth benches are small- n (bootstrap CI on the three-hop flat arm $\approx \pm 0.30$). The zero-LLM baseline is strong on one-relation analogies (0.641), which makes the terrain honest for the nulls but weak for showcasing the prior; the depth results are where the prior’s contribution is cleanly positive.
- **Exact verifier in the stress test.** The 0.000 $\rightarrow$ 1.000 repair uses an exact stored relation table as the verifier. In a live agent the verifier is a lived, noisy graph; the result shows who *owns* the correction, not that correction is free.
- **Held-out injection is simulated namelessness.** The nameless case holds task concepts out of the bridge’s training vocabulary; a live nameless percept would come with perceptual context we do not model. The information-theoretic reading (meaning does not determine spelling) is indifferent to this, but the simulation should be kept in mind.
- **The in-language demarcation rests on two worlds.** Program search (pays) and one text adventure (inert) instantiate the predictable-from-language axis; more worlds along that axis would firm the boundary. The rented-vs-owned duel is also on the encoder-favourable side (specs close to string matching), a bound we state rather than hide.
- **Deployment arms are single-run.** The search benches are deterministic given (world, seed) and reported as counts (16/16, 73 rejections), not CIs; the analogy and dose-response benches carry the statistical weight.

11 Conclusion

There is no latent shortcut into a frozen LLM’s reasoning prior. The prior is real (token CoT 0.859 against a chance-level weight-shuffled control), but it unlocks through the dis-

Table 3: The rented prior at deployment grain (benches PROPOSER-DOMAINS and RENTED-VS-OWNED). One propose-verify search; the world’s verifier owns correctness on every arm (zero garbage committed). The prior pays only in-language, and the owned 22M encoder matches the rented 1.5B model there.

world	proposer	solved / covered	verifications	rejections	reading
NL→program (16 goals, budget 40)	rented LLM	16/16	3.7	—	pays: $3.5\times$ fewer, $5\times$ coverage
	random	3/16	13.0	—	floor
Colossal Cave (budget 140)	rented LLM	18 rooms	140	73	inert: prose under-determines exits
	random	17 rooms	140	73	identical rejections
NL→program (24 goals, budget 40)	rented LLM (1.5B, ~ 3 GB)	24/24	3.1	—	
	owned encoder (22M, ~ 80 MB)	24/24	2.1	—	matches the rented prior
	random	11/24	23.0	—	floor

crete autoregressive selection, not through any continuous read of the model’s state (best continuous read 0.192, degrading with depth), and it admits input only through token identity, not meaning (nameless latents at chance held out, 0.031, even for a converged functionally trained projector, 0.078 ± 0.064 ; name tokens at 0.812). This double lock is not an engineering failure but a property of how the model’s manifold is organised, and it freezes a sane interface: grounding (admit, then *name*) is not a step to shortcut; the name is the bridge. Depth then belongs to the agent, not the model: recursing through the concept inventory carries what continuous chaining destroys (0.900 vs. 0.100), pays accuracy at three hops (1.000 vs. 0.750), and puts a verifier at every hop, which is where correction lives (0.000 → 1.000 under injected hallucination). As for what the rented prior buys once plugged in correctly: in-language ordering efficiency ($3.5\times$ on program search), nothing where only interaction knows (a real text adventure), nothing decisive that a $40\times$ smaller owned encoder does not already do (2.1 vs. 3.1 verifications), and never the correction; its confidence, confidently wrong at $p_{\text{top}} = 0.971$, earns no commit rights. The open questions are stated as questions: does the double lock replicate across frozen models? where exactly does the in-language boundary run, between a spec that matches strings and one that requires the prior’s depth? and can a naming faculty be made cheap enough that the bridge is never the bottleneck?

Reproducibility statement

A self-contained code supplement (the `code/` directory accompanying this paper) reproduces every number: public dependencies only (`torch`, `transformers`, `sentence-transformers`, `scikit-learn`, `numpy`, and the public-domain adventure package for the real game), no private code, both frozen models fetched once from the public hub. Every bench script prints its pre-registered hypotheses, its controls, and a computed verdict; all randomness is seeded and explicit; quads and worlds regenerate bit-for-bit from (configuration, seed). Appendix A maps every bench name used in the text to its script.

References

- Belnap, N. D. A useful four-valued logic. In Dunn, J. M. and Epstein, G. (eds.), *Modern Uses of Multiple-Valued Logic*, pp. 5–37. D. Reidel, Dordrecht, 1977.
- Belrose, N., Furman, Z., Smith, L., Halawi, D., Ostrovsky, I., McKinney, L., Biderman, S., and Steinhardt, J. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.
- Deng, Y., Prasad, K., Fernandez, R., Smolensky, P., Chaudhary, V., and Shieber, S. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*, 2023.
- Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., Grosse, R., McCandlish, S., Kaplan, J., Amodei, D., Wattenberg, M., and Olah, C. Toy models of superposition. *Transformer Circuits Thread*, 2022.
- Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, 2021.
- Ghandeharioun, A., Caciularu, A., Pearce, A., Dixon, L., and Geva, M. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 15466–15490, 2024.
- Goyal, S., Ji, Z., Rawat, A. S., Menon, A. K., Kumar, S., and Nagarajan, V. Think before you speak: Training language models with pause tokens. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.
- Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

- Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., and Kaplan, J. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.
- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems 35*, 2022.
- Lester, B., Al-Rfou, R., and Constant, N. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.
- Li, X. L. and Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pp. 4582–4597, 2021.
- Merrill, W. and Sabharwal, A. The expressive power of transformers with chain of thought. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.
- Mikolov, T., Yih, W.-t., and Zweig, G. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746–751, 2013.
- nostalgebraist. Interpreting GPT: The logit lens. LessWrong, <https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens>, 2020.
- Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., and Wu, X. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599, 2024.
- Pearl, J. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.
- Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- Reimers, N. and Gurevych, I. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 3982–3992, 2019.
- Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems 36*, 2023.
- Subramani, N., Suresh, N., and Peters, M. E. Extracting latent steering vectors from pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 566–581, 2022.
- Turner, A. M., Thiergart, L., Udell, D., Leech, G., Mini, U., and MacDiarmid, M. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.
- Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems 33*, 2020.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35*, 2022.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. ReAct: Synergizing reasoning and acting in language models. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.
- Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., Goel, S., Li, N., Byun, M. J., Wang, Z., Mallen, A., Basart, S., Koyejo, S., Song, D., Fredrikson, M., Kolter, J. Z., and Hendrycks, D. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Bench index

Table 4: Bench names used in the text, their role, their substrate, and the reproducing script in the code supplement.

Bench	Role in this paper	Substrate	Script
LATENT-BRIDGE	latent→latent reflection null; loss decomposition; shuffled control	64 analogies, 8 relations	bench_latent_bridge.py
ZERO-TOKEN-READS	soft-distribution and latent-autoregression nulls	64 analogies	bench_zero_token_reads.py
INVENTORY-SCORING	sharp selection over the concept inventory; injection gap	64 analogies	bench_inventory_scoring.py
GATED-DEPTH	2-hop: discrete collapse vs. continuous intermediate; audits	10-country relation table	bench_gated_depth.py
DEPTH3-VERIFICATION	3-hop accuracy payoff; hallucination stress, verify ON/OFF	8-row monument chain	bench_depth3_verification.py
INJECTION-DECOMPOSITION	mono-vector collapse vs. linear shrinkage vs. name tokens	64 analogies	bench_injection_decomposition.py
NAMELESS-INJECTION	held-out (nameless) injection null, linear bridges	64 analogies, 88 held-out concepts	bench_nameless_injection.py
LEARNED-PROJECTOR	functionally distilled MLP projector, held-out null	64 analogies, 88 held-out concepts	bench_learned_projector.py
PROPOSER-DOMAINS	where the rented proposer pays: in-language vs. interaction	Colossal Cave + NL→program	bench_proposer_domains.py
RENTED-VS-OWNED	1.5B rented vs. 22M owned proposer, equal budget	NL→program (144 programs)	bench_rented_vs_owned.py
CONFIDENCE-GATE	dose-response of confidence-gated commits	12 unverified capital items	bench_confidence_gate.py

B The canonical interface, spelled out

The interface measured piecewise in Sections 4–7 composes into one loop, entirely outside the frozen model:

1. **Perceive.** A percept is encoded by the agent’s frozen encoder into the 384-d latent.
2. **Admit.** The gate decides whether the latent enters the concept inventory; the gate margin separates in-inventory from out-of-inventory queries ($\Delta = +8.68$ measured in bench GATED-DEPTH), giving a calibratable abstention signal.
3. **Name.** An admitted concept receives a stable token string. This step is the bridge: without it, no channel into the frozen model generalises (Section 7).
4. **Inject.** The name’s real sub-token embeddings enter the frozen model’s context ($0.812 \approx$ token CoT; the mono-vector average of the same tokens costs -0.312).
5. **Reflect and select.** The frozen model scores every inventory candidate (teacher-forced, length-normalised log-probability; one forward per candidate; zero tokens emitted); the argmax selects. Continuous alternatives measurably fail (Section 5).
6. **Verify.** The selected edge is checked by the agent’s own verifier (a stored relation, a probe on a world copy, an execution). Agreement files as True; contradiction is retained as Belnap’s Both and triggers re-selection among touch-consistent candidates ($0.000 \rightarrow 1.000$ under injected corruption) or abstention.
7. **Recurse.** The selected concept’s *name* seeds the next hop’s prompt; depth is the number of gated hops, not a property of the model’s continuous state (0.900 vs. 0.100 ; three hops 1.000 vs. 0.750 flat).

Two dials govern the loop’s honesty. Selection temperature (default 0: argmax, the regime every measurement here supports). And commit rights h for unverified hops: a hop with no verifiable edge is always retained as Both; it may be committed only as a *labelled hypothesis* and only if $p_{\text{top}} \geq 1 - h$. Bench CONFIDENCE-GATE measures why $h=0$ is the only safe default: the dose-response is flat and the model is confidently wrong exactly where the agent’s lived world contradicts the prior.