

Trace AI — The Structural Intelligence Program

The Structural Intelligence Program

0. The whole thing in one paragraph
1. The conjecture, stated simply
2. Trace AI is the conjecture applied to cognition
3. Three build directions — each is a product surface
 - 3.1 Representation search → **the schema is the answer to “what do we keep?”**
 - 3.2 The compiler → **the corpus proves one structure, many substrates**
 - 3.3 Symbolic-causation measurement → **the eval suite is the product a lab buys**
4. What this changes for the build (three small deltas, no new pillars)
5. What we are *not* claiming (kept explicit, on purpose)
The one-line version, for the deck

The Structural Intelligence Program

Trace AI research note, v0.1 — 2026-08-03. One job: take the three “constructive projects” from the structure/selection thread and say, as plainly as possible, what each one *is* when it is pointed at Trace AI as a product. Companion to `AGENCY_AND_SELECTION.md` ; both develop the ideas the whitepaper keeps out of its body.

0. The whole thing in one paragraph

Intelligence, the thread claims, is not mainly *predicting the next state or picking the next action*. It is finding the **representation** in which a problem becomes at once **compressible** (you can throw away most of the detail) and **controllable** (small, legible changes to what you kept move the outcome the way you intend). Trace AI is that claim, aimed at one specific system: a mind in the middle of thinking. The self-representation r_t is the compressible-and-controllable representation of a cognitive process. The schema says what to keep. The corpus proves the same structure survives across different substrates. The governance eval measures whether

the thing you kept actually controls the outcome. Everything below is those three sentences, one per section.

1. The conjecture, stated simply

Structural Intelligence Conjecture. A capable system’s core move is to find a map $q: X \rightarrow Z$ from a messy space X to a representation Z such that (1) the outcome-relevant dynamics still make sense in Z , (2) everything irrelevant is thrown into the parts of X that q collapses, (3) useful interventions can be written down *compactly* in Z , and (4) those interventions keep working when you move to a new situation.

Plain version: **intelligence finds the level at which the world becomes both simple to describe and easy to steer.** Too much detail and you can’t act; too little and you lose control. The skill is finding the middle.

This is not new mysticism. It is what a good coordinate system does for a physics problem, what a good abstraction does for a program, what a good diagram does for a proof. The thread’s contribution is to notice it is the *same* move every time, and to ask whether a machine can search for that move directly.

2. Trace AI is the conjecture applied to cognition

Line up the conjecture with the whitepaper and it is the same object:

Conjecture	Trace AI
messy space X	the model’s full compute state z_t (millions of activations)
representation Z	the self-representation r_t — a few legible slots
the map $q: X \rightarrow Z$	the decoder $h_\phi: z_t \mapsto r_t$
“outcome-relevant dynamics survive in Z ”	intervening on r_t moves the next action ($\mathcal{L}_{\text{gov}}$)
“irrelevant detail is collapsed”	r_t is a <i>bottleneck</i> : what doesn’t fit doesn’t reach the action

Conjecture	Trace AI
"interventions are compact and legible"	r_t is human-readable by constraint (whitepaper §2.1)
"keeps working in new situations"	the transfer criterion (AGENCY_AND_SELECTION.md §4, criterion 6)

So Trace AI is not *inspired by* the conjecture; it is the conjecture with the messy space fixed to "a mind mid-thought" and the extra demand that the representation be **legible to a human**, not merely small. That extra demand is the product. A tiny illegible bottleneck would satisfy the compression half and fail the point. Trace AI wants the level that is compressible, controllable, *and* readable.

3. Three build directions — each is a product surface

The thread proposed three projects. Read for Trace AI, each is not a new company — it is a thing Trace AI already half-is, named clearly enough to build on purpose.

3.1 Representation search → the schema is the answer to "what do we keep?"

The first project imagines a machine that searches over candidate maps q for the one that makes a hard problem simple. For Trace AI the search target is fixed and concrete: *which slots does r_t need?* Goal, belief-about-task, belief-about-self, uncertainty, planned-next — that list is a hypothesis about which distinctions in a thinking process can be safely collapsed and which must be preserved to keep the outcome controllable.

- The **schema** (schema/reasoning_trace.schema.json) is the current best guess at that representation.
- The rule "the schema follows the data — do not add a slot without a session that needs it" is *representation search done by hand*: each new trace that the vocabulary cannot capture is evidence the current q collapsed a distinction it shouldn't have.
- The **governance eval** is the test of the search: if intervening on the kept slots doesn't move action like a human's would, the representation is missing something.

Product consequence, simply: the moat is not the model, it is the *right list of what to keep*, discovered from real cognition and validated by intervention. Every trace that forces a schema change is the search taking a step. Automating that step — proposing slot changes from traces

the vocabulary strains against — is the first place “representation search” becomes a tool inside the pipeline rather than a metaphor.

Where the search has provable teeth. The SIC formal paper

(`SIC_MATHEMATICAL_FOUNDATIONS.md`) makes representation-search a *theorem* — the minimal-sufficient partition of Halmos–Savage exists for any well-posed task (Theorem 1), and cross-task stability holds iff the task family admits a shared Markov screen (Theorem 4). What used to be “we discover the slots by hand-tuning against traces” is now: the slots are the atoms of a minimal-sufficient σ -algebra over the counterfactual task family, and any two slot sets that both satisfy the sufficiency condition against the same task family are equivalent up to a q -preserving relabeling. **And with the right inductive bias — statistically independent, non-Gaussian slot components — Theorem 7 gives polynomial-in- d_r recovery** (classical linear ICA; Instrument 8 witnesses it). That is the mathematical form of “we can afford more slots without paying an exponential collection cost, if the bottleneck is engineered for identifiability.”

3.2 The compiler → the corpus proves one structure, many substrates

The second project wants a “compiler” that takes one substrate-independent structure and instantiates it in many materials, with tests showing the same organization survives. Trace AI already has the beginnings of this, and it is the reason the data is defensible.

- **A trace is the structure; a subject is a compiler.** The schema specifies an organization (perception → recall → uncertainty → correction → action) without specifying a substrate.
- The reference corpus already instantiates the *same* organization across substrates: Session 7f3a (human, typed), 7f3b (human, spoken), 7f3c (model, text). The `reaffer` → `re-affir` → `reaffirm` correction chain is literally one structure surviving three instantiations across two subject kinds — a worked example of substrate-independent structure with the diagrams commuting.

Product consequence, simply: because the training signal is a *structure*, not a model’s internal format, it is not tied to any one architecture — it re-compiles onto whatever model you train next. That is what makes the corpus an asset that outlives a model generation, the same way a dataset outlives the network trained on it. The capture app, the human annotator, and an instrumented model are three compilers for the same schema; the product is the schema plus the compilers, and the corpus is the proof they agree.

3.3 Symbolic-causation measurement → the eval suite is the product a lab buys

The third project wants to turn “symbolic structures change the world” into a measurable science: introduce a symbolic structure m (a prompt, a rule, a self-model), and measure how much it reorganizes the distribution over futures,

$$\Delta_m = D(P(\gamma \mid s_t, m) \parallel P(\gamma \mid s_t)),$$

then separate mere *influence* from real *control*, *knowledge*, and *agency*. This is exactly what Trace AI’s governance eval and functional-aliveness profile already do, with $m = r_t$. The self-representation is the symbolic structure; $\mathcal{L}_{\text{gov}}$ is the measurement of Δ_{m_i} ; the six criteria in `AGENCY_AND_SELECTION.md` separate influence from control from agency.

Product consequence, simply: the eval suite is not internal QA — it *is* a product. An alignment lab does not primarily want another model; it wants a **reproducible instrument** that says how much a model’s stated reasoning actually governs its actions, how well it is grounded, and whether it correctly credits its own effects. That instrument, run on *their* model as well as ours, is the “Symbolic Causation and Agency Benchmark” — and it is the thing that turns “faithful reasoning” from a slogan into a number a partner can put in a report.

4. What this changes for the build (three small deltas, no new pillars)

Adopting the frame does not add a new product; it renames and sharpens what exists.

1. **Schema work is representation search.** When a trace strains the vocabulary, log it as a *search step*, not a chore. The medium-term tool is an assistant that reads strained traces and proposes slot changes — the schema’s own gradient.
2. **The corpus’s cross-substrate agreement is a headline, not a footnote.** “The same reasoning structure compiles onto humans and models and the diagrams commute” is a stronger claim to a research partner than “we have traces.” Lead with it.
3. **Package the evals as a standalone benchmark.** The governance eval + the six-criterion profile, runnable against any checkpoint, is the artifact a lab can adopt before it ever licenses a model. Ship it as a spec first (it costs a document), a harness second.

5. What we are *not* claiming (kept explicit, on purpose)

The same discipline as the whitepaper and `AGENCY_AND_SELECTION.md` :

- Not that Trace AI has *built* a representation-search engine, a universal compiler, or a finished benchmark. It has the pieces and the frame that says how they fit.
 - Not that finding a compressible-and-controllable self-representation produces consciousness. That question is bracketed (whitepaper §3.2; `AGENCY_AND_SELECTION.md` §5); the profile is an instrument for asking it, not an answer.
 - Not that the conjecture is proven. It is a lens that happens to describe Trace AI cleanly and suggests where to push. Its value is that it turns “find the latent space between these ideas” from a poetic instruction into concrete questions — *what do we keep, what compiles, what controls* — that the product already answers in miniature.
-

The one-line version, for the deck

Trace AI finds the level at which a mind’s thinking becomes small enough to read and structured enough to steer — the schema says what to keep, the corpus proves it compiles across humans and models, and the eval measures whether what we kept actually controls the outcome.

v0.1, 2026-08-03. See `AGENCY_AND_SELECTION.md` for the selection/agency development and `DECISIONS.md` (D23–D24) for provenance.