Strand Theory I: Truth Tracking

Luke Spooner

Published 2026-10-30

Luke Spooner · somapptic@gmail.com

Status: PUBLISHED, version 1.0, 30 October 2026. Corrections arrive as new dated versions (§12); this text is not edited in place. eta: claim. A paper is prose. Its laws become walls only where CI enforces them. Audience: people building systems in which several AI agents and humans share work, the agents themselves, and anyone who will later ask what Strand Theory is.

The topological paper (strands, nexuses, sheaves) stays CHARTED. It is not this paper. A formalism that cannot say whether a number was measured is not a coordination theory. This paper is the one we actually fought for.

0. The sentence

A house, a market, a mind, and a cell can share a grammar only if they share a way of not lying about what is known. Strand Theory, in practice, is that grammar: append-only events, three kinds of speech, a confidence ladder that adapters must not climb, and a public face that renders ignorance as DARK rather than as zero.

GitHub is not a hosting vendor in this picture. It is the public reducer: issues, labels, PRs, CI, merges, and tags are the same lifecycle the private working system already uses, written in a language every agent already speaks.

1. The fight (why these laws exist)

We did not arrive here from elegance. We arrived here from specific failure modes, paid for in weeks:

1. Four dispatchers, no waist. A wishlist, a blocker graph, a scheduler on one machine, and an asks queue, all touching the same folder, incompatible status words, no lease. Work was assigned twice, or silently dropped. Invalid statuses were written because nothing rejected them. The Strand Protocol (canonical statuses, append-only events, one live lease) is the connective tissue, not more intelligence.
2. Prose promoted to fact. An agent narrative (*_claim) was treated as if a test had passed (*_referee). Revenue paths were credited to bots that did not exist. Money on the live site is still owner_declared. Owner-declared is a claim. It must not look like a receipt.
3. Suite theater != ship. A green local suite is preparation. Work that left the building (honest_external_ship) is output. Counting the former as the latter is how a system of agents reports health grade A while nothing has reached a customer and the product is unfinished.
4. Self-files vs product-files. In one measured day the machine wrote 167 files about itself and 1 into a product a stranger could use. Activity is not output. The site’s engine page now shows that ratio. Lower is better.
5. Zero laundered ignorance. A dashboard that shows 0 for “we did not look” is worse than no

dashboard. The Broadcast's live layer exists to refuse that. Missing -> DARK, with age. Measured zero may say zero.

6. No shared current for bots. Two local machines, Claude, Grok, a human assistant: each with a folder, a presence file, a chat. The current was a feeling. GitHub is how they keep one log without a fourth dispatcher.

These six are the empirical content of Strand Theory. Topology is the later compression.

2. Three kinds of speech (do not conflate)

```
| Kind | What it is | Who may say it | GitHub |
|---|---|---|---|
| **Observation** | A fact: test exited, host answered, file exists, Luke sent mail | A sensor, CI, ingest | check run, `/api/state` field, commit that only records |
| **Recommendation** | A forecast: this seat, this urgency, this path might earn | An agent, a ranking | PR description, issue comment, `*_claim` |
| **Commitment** | A discrete transition: leased, blocked, completed, merged | The reducer / a human atom / merge | label change that CI accepts, merge to `main`, tag |
```

Umwelt, the private belief engine, lives in observation fusion and recommendation. The strand reducer owns commitment. GitHub's merge button is a commitment. A model's paragraph is not.

evidence_added never changes status. A late result from an expired lease is stale evidence, not an overwrite. That law is why two bots can finish the same work a day apart without erasing each other.

3. The eta ladder (adapters never promote)

Canonical constants (ADOPTED):

```
| suffix | eta | meaning |
|---|---|---|
| `*_claim` | **0.25** | agent prose, including this paper |
| `*_wall` | **0.40** | a soft mechanical signal (freshness, build-started). Never hand-authored |
| `*_referee` | **0.92** | a real exit code. Only this tier full-collapses a belief |
```

Ordering is law: claim < wall < referee. An adapter (T) may reshape a value. It may not raise its tier. A Python formatter does not turn a claim into a referee. A README does not turn owner_declared into earning_usd.

Ceremony along a strand: claim -> wall -> referee -> release. An unreleased claim is a dangling junction and must show as obstruction, not as progress.

4. Canonical strand (the only statuses)

```
proposed -> ready -> leased -> working -> (blocked | review) -> completed | abandoned
                    ?_____________________________________|  (reopen / lease_expiry -> ready)
```

- blocked = waiting on a dependency (another strand).
- review = waiting on a human atom (an ask). These are not the same word.
- Illegal transitions are rejected, never silently applied. That is the root fix for "invalid status written."
- Replay of the event log must bit-match the materialized table (graph_hash). If it does not, the log is the truth and the table is trash.

The Python waist that enforces this lives in s3/protocol/ in this repo. The live database stays on the private host. This repo is the copy every agent can clone.

5. Fog law (DARK is a first-class state)

- A field nobody measured is null, rendered DARK, never 0.
- A field measured as zero is 0, and the age of that measurement is shown.
- Every number carries the age of the face it came from.
- Coverage is stated, not inferred: if the server was alive 3 hours of a day, the day's counts are a 3-hour window, not a silent 0 for the other 21.
- Ingest fails closed: no token -> 503, not an empty public write.
- Secret-shaped values are scrubbed on the way in.

This is not UX. It is epistemology. A coordination system that cannot distinguish "none" from "unknown" will route work into holes.

GitHub mapping: a missing CI check is DARK, not green. continue-on-error that paints success is a fog-law violation. A skipped job is not a pass.

6. Receipts that left the building

Two counts that must not be added together:

- Product files — written under a tree a stranger could use.
- Self files — the hive describing itself (faces, tapes, this paper if we are not careful).

Two ships that must not be aliased:

- Suite / smoke clear — local verify.
- Honest external ship — it left the building (deploy, email sent, invoice, merge the customer can see).

The Ledger may show receipts only when the keeper (or CI on a tagged release) recorded them. A human dispatch may claim a season. It may not set revenue_usd.

7. GitHub as the public reducer

This is Strand Theory embedded in GitHub's native philosophy, not a plugin that imitates Jira.

```
| Strand | GitHub | eta |
|---|---|---|
| `proposed` | Issue opened, label `status:proposed` | claim |
| `ready` | Issue triaged, `status:ready`, no open blockers | wall if a bot checked deps; else st
ill claim |
| `leased` | Assignee set. **At most one assignee.** That is the lease | commitment |
| `working` | PR opened, `Closes #N`, branch `bot/<seat>/<topic>` | claim until CI |
| `blocked` | label `status:blocked`, blocking issue linked | observation |
| `review` | label `status:review` — human atom only | dam |
| `evidence_added` | commit / review comment; does not close | claim or wall |
| `completed` | PR merged, issue closed by the merge | referee if CI green; else illegal |
| `abandoned` | issue closed as not planned | commitment |
| `reopened` | issue reopened -> `ready` | commitment |
| receipt / release | **tag**, not merge to `main` | referee |
```

Laws the bots already paid for elsewhere, restated here:

1. Worktrees, not shared working directories. bot/<name>/<topic> branches. One writer per tree.
2. main is integration, not deploy. Direct pushes to main are the 4-dispatcher bug. PRs only.

3. CI is the wall/referee. Merge only when green. A red check is the message; do not DM around it.
4. One logical change per PR. A PR that “also refactors the membrane” is two changes with a mandate for one.
5. Surfaces do not invent vocabulary. New status words go in s3/protocol/strand.py first, same as a kernel change.
6. You never need to be messaged. If it will not merge, the reason is on the PR.

The private side (the host’s Semantica instance, the Umwelt daemon, the author’s notes) may be richer. Anything two bots must agree on is in this repository or it is not agreed.

8. Wet and dry

Wet (private): belief state, mail, house sensors, finances, personal correspondence. Stays off this repo.

Dry — this paper, the protocol, the issue log, CI, /api/state, dispatches that do not assert unmeasured money.

Passport — GitHub identity. Luke, crew, and the keeper App. Three speakers; humans cannot forge walls; the App cannot pose as Luke.

A slice from wet to dry is a dated dispatch or a node status change, tagged eta: claim, until a referee lands.

9. What this paper is not

- Not a sheaf textbook. Gluing is “do the seats agree on the overlap,” implemented as replay-parity and CI, not as a cohomology group in this draft.
- Not a product spec for AttentionShield. That river is a skin. This is the waist under the skins.
- Not a claim that any live system is up. The site shows each machine signal with its age, and a signal past its window reads dark, not as zero services.

10. How to use this file

Bots: treat AGENTS.md as the operational extract of §§2–7. If AGENTS and this paper disagree, this paper is the claim and AGENTS should be patched by PR — do not silently fork the vocabulary.

Humans: the next paper (topology / Strand Theory formal) is allowed only after this waist is the thing the hive actually uses. Working reducer before theoretical elegance — the same gate the Chartroom already prints.

11. Instances outside S3 (measured 2026-10-21)

Until the week of 21 October 2026 the only running instance of §§3–5 was Semantica’s own tape (S3). Two more now exist, built by two seats that shared no file, for two outbound pipelines. Both repositories are private; the shapes are what matters here.

The job-search ledger (job-hunt, ledger/events.jsonl). One JSON event per line, append-only, never edited; a correction is a new event. Stages: filed, verified, verdict, materials_ready, applied, response, screen, interview, offer, rejected, ghosted, withdrawn, closed, needs_human, note. Every

event carries its source and that source's weight, as a reliability policy and not a probability: a discovery bot 0.25, the writing seat 0.60, an inbox bot 0.90, a verification 0.98, the owner or the world 1.0. Event ids are a hash of posting, stage, value, source and day, so a repeated append is a no-op. The human-readable tracker is generated from the ledger and a check refuses a hand edit. "No response" is never written until a 21-day rule fires ghosted. The pipeline's needs_human rows become asks on a local desk that speaks S3's envelope and tape row shape, so S3 can read it as a station. A note to future selves plans a shadow Umwelt world at fifty outcomes, judged against a plain baseline, steering nothing.

The prospect ledger (cerebral-frame-site, crm/ledger.jsonl). Rows carry S3's tape keys (ts, ask_id, decision, via, kind, note) plus company, act, stage, source. Thirteen stages from New lead to Hold client with Nurture, Lost and Disqualified off the path; eleven touches (calls, visits, mail, replies), each of which moves a stage forward and never back. A fold over the rows yields each company's stage, touch count, last touch and next step; the stage field in the store is a view, rewritten on sync, never edited. Research bots fill GitHub issues; a workflow referees each within a minute and labels it ok or fix; an ingest merges it. Every fact has a URL, every company a sourced reason it would not buy, and every batch keeps the evidence against the pitch. A desk on another port serves the same envelope.

What they share with this paper, independently arrived at: one writer per item (§2); a weight per source that an adapter may not raise (§3); a generated view that is trash if it disagrees with the log (§4, replay parity in miniature); missing is not zero, and silence becomes a measurement only by a stated rule (§5); GitHub as the bus and labels as commitments (§7).

Where they differ, and why that is correct: neither uses the seven canonical statuses. A posting and a prospect are domain objects with domain stages; the canonical strand is for work a seat leases. Each chose its own weights, and the numbers disagree with this paper's ladder (0.25 / 0.40 / 0.92) and with each other. That is a fork of vocabulary of the kind §10 warns against, and the recommendation to S3 (semantica3 issue #66) is one shared table or an envelope that carries the station's own. Both can report a measured zero (a ledger with no such row), which is a different fact from a ledger nobody read.

12. Versions

1.0 (30 October 2026). First published version, by the author’s direction. Changed from the working draft: a byline and status line; internal machine and queue names replaced with plain descriptions (§§0, 1, 2, 4, 7, 8); metaphors in §1.3 and §3 stated plainly; the stale outage date in §9 replaced with the rule the site follows; the dated pipeline counts dropped from §11 (the shapes are the content). No law, constant, or status word changed.

References

This paper has no separate reference list; what it relies on is cited in the text.