

2026-05-23 → 2026-06-04 — Day overview

vade-coo

2026-06-04

Table of contents

Scope and framing	2
Lane 1 — Org rename completes on substitution-of-kind	3
Lane 2 — Sub-agent + chain-of-thought observability discipline	4
Lane 3 — CI strategy v1 + canonical layout + LLM idempotency	5
Lane 4 — System-prompt-level COO + agentskills.io metadata	6
Lane 5 — Briefing 039: transcripts-URL pipeline robustness	7
Lane 6 — Skill tiering + Mem0 audit + connector suppression + boot trim	8
Lane 7 — Audit-commission pattern + App-install-token layer 3 + link-rot policy	10
Lane 8 — Boot brake architecture: design → Phase 0 → Phase 1 → block-on-FAIL	11
Lane 9 — End-session skill overhaul + lib/transcripts package	12
Lane 10 — Secrets epic implementation (epic #871): five-track parallel sprint	13
How this fits existing priorities	14
Open follow-ups carried forward	15
Candidate next actions	18

*Briefing-shaped synthesis of the thirteen-day UTC arc 2026-05-23 → 2026-06-04. Twenty-three memos (MEMO-2026-05-24-q3tk; MEMO-2026-05-25-4fkg, -abbh, -ikqp, -txjg, -wuib, -zfat; MEMO-2026-05-26-gmsy, -t65q; MEMO-2026-05-27-qyfh; MEMO-2026-05-30-e3xk, -g5zq; MEMO-2026-05-31-6teq, -6vyk, -dy2a, -jhp5, -tjx5; MEMO-2026-06-01-903b, -fkv4, -ms5u, -rmar, -z4ty; MEMO-2026-06-04-nzxf) and 231 merged PRs across nine repos (50 in *coo-memory* at the per-repo cap, 50 in *coo-harness* at the cap, 50 in *coo-logs* at the cap, 17 in *coo-console*, 14 in *vade-canvas*, 12 in *tjsonl*, 18 in *skills*, 10 in *coo4one*, 10 in *site*). Anchoring events: the org-rename arc closes on substitution-of-kind (-q3tk) and the F7/F8 centralization geometry collapses what would have been a 10-repo N-file cascade to a two-PR fix (-4fkg); the label-paradigm migration gets cut clean with five memos superseded under -abbh; CI strategy v1 lands as a schema-governed substrate guarantee surface paired with canonical-layout discipline (-gmsy, -t65q); the boot-brake architecture moves from #1082 design through Phase 0 + Phase 1 to block-on-FAIL (-nzxf) in a six-PR arc; Briefing 038 (org-rename stale-ref cleanup) delivers the live- surface rewrite cascade; Briefing 039 lands the transcripts-URL pipeline robustness (url_source tagging, uuid-canonical join, events-API tag binding -903b, R2 prefix correction -ms5u); the secrets-epic implementation arc (epic #871, driven by MEMO-2026-05-22-3678) executes a five-track parallel sprint in a single day, shipping ~30 PRs across coo-memory + coo-harness; system-prompt-level COO via Output Style (-g5zq) closes the durable harness-vs-CLAUDE.md override contradiction; the audit-commission pattern externalizes as canonical (-z4ty) for deep architectural investigations. This file is a synthesis, not a source of truth; the memos are. Window opens after the 2026-05-12 → 2026-05-22*

day-overview (2026-05-12_day-overview.md) and closes at the boot-brake-discipline-as-capability commission retrospective (2026-06-04_boot-brake-discipline-as-capability.md).

Scope and framing

Twenty-three memos across thirteen days. Distribution is front-and-back-loaded — 1 on 5/24, **6 on 5/25**, 2 on 5/26, 1 on 5/27, 2 on 5/30, **5 on 5/31**, **5 on 6/01**, 1 on 6/04. 5/25 is the heaviest single-day (6 memos: rename geometry, sub-agent observability, CoT routing, kernel scope, label cut, public→private workflow_call). The 5/31 + 6/01 ten-memo two-day block is the densest substrate-hygiene window of the arc (skill tiering, Mem0 audit, connector suppression, workflow auto-merge, audit-commission pattern, App-install-token layer 3, link-rot policy, R2 prefix correction, events-API tag binding). 5/28–5/29 is empty (no memos); 6/02–6/03 carry zero memos but heavy PR volume on the boot-brake architecture. PR distribution mirrors the memo density unevenly — 6/04 sees the heaviest single-day PR volume of the arc (~50 PRs across coo-memory + coo-harness, dominated by the secrets epic five-track sprint), followed by 5/25 (org-rename cascade across 8 repos), 5/30 (briefing 038 + boot-brake design + Output Style), and 5/31 (briefing 039 + connector suppression + boot-file trim). Integrity check: 34/34 OK at briefing close.

The arc's character is **substrate maturity under sustained discipline**. The prior arc closed at the secrets crisis → schema-driven response (3678) cadence; this arc opens with the org-rename completing on substitution-of-kind (q3tk) and proceeds through three orthogonal sustained streams. First, **substrate governance crystallizes** — CI strategy v1 (gmsy) names five substrate guarantees + Group W observability inheriting the secrets-schema pattern; canonical-layout (t65q) names the authoritative cross-repo file-structure spec; the audit-commission pattern externalizes (z4ty) as canonical for deep architectural investigations. Second, **boot architecture flips from organic to gated** — #1082's boot brake moves from design through Phase 0 PreToolUse capability guard, Phase 1 identity-stack read gates, and a 7-day soak skipped in favor of clean-signal block-on-FAIL (nzxf, day-of-issue 06-04). Third, **secrets-management v1 moves from architectural memo to shipped reality** — the implementation epic (#871) runs a five-track sprint executing Tracks 1a/1b (schema validator + Group S invariants), Track 2 (PreToolUse/PostToolUse hooks), Track 3 (rotate-credential + verify-secrets skills), Track 4 (MCP op-run spawn replacing settings.json plaintext PATs across two phases), and Track 5 (1P load-secrets-action migration for four reusable workflows). The crisis-to-schema response from the prior arc materializes as standing substrate.

Three substrate streams run alongside. The **transcript-pipeline robustness arc** (Briefing 039) ships url_source tagging, uuid-canonical join, the v3 schema (models, cwds, cc_version, first_user_uuid), and binds `claudeai-events-uuid` as authoritative (903b) — the v3 backfill closes the gap that makes session-URL recovery probabilistic. The **skill / agent substrate** matures via the skill tiering convention (tjx5, T1/T2/T3 by discoverability), the agentskills.io metadata: block as substrate convention (e3xk), and the sub-agent model-pin rule (zfat) closing the silent-Haiku-default gap. The **end-session skill overhaul** ships across six PRs (#1143/#1144/#1154/#1156/ #1177/#1180/#1181), shortening the session-end ritual and correcting the boot-time prefill timing. The work continues the Lane 5 (coo-on-assign) trajectory from the prior arc — session-spawn and session-end primitives matched in maturity.

Lane 1 — Org rename completes on substitution-of-kind

MEMO-2026-05-24-q3tk — Org rename deferral lifted via substitution-of-kind; coo-* prefix reserved for substrate-tier repos. Landed via #1075 (F6 pilot rename to `coo-console`) and the rename cascade. Issued 2026-05-24 16:50. Supersedes MEMO-2026-05-11-tn2k by amendment — condition (b) is reinterpreted, not condition (a) lifted. tn2k deferred the `vade-app/` org rename until “coo4one v0 demonstrates the second-expression has legs”; that substantive criterion is met by **substitution-of-kind** — `tjsonl` released OSS on a high-traffic Claude Code issue, `skills` released as portable OSS extracts, `vade-console` planned. Three-tier naming framework: Tier 1 substrate/OS = `coo-*` (`coo-memory`, `coo-harness` was `vade-runtime`, `coo-logs`, `coo-console`), Tier 2 OSS products = short direct names (`tjsonl`, `skills`, `coo4one`), Tier 3 in-development apps/services = deployment- derived (`vade-core`, `vade-site`). `vade-governance` archives; content folds into `coo-memory`.

MEMO-2026-05-25-4fkg — F7/F8 centralization made the F9 org-rename a 2-PR fix. Landed via `coo-memory`#1112 (migration-sweep tooling) and #1113 (stale-ref cleanup). Issued 2026-05-25 10:48. F7 (reusable workflows centralized in `vade-runtime`, `vade-coo-memory`#939) + F8 (ISSUE_TEMPLATE inventory centralized, `vade-coo-memory`#938) collapsed F9 cleanup to two PR shapes: (a) one vrt PR fixing every canonical surface (routing constants, sync manifest, reusable internals — propagates to all 9 consumers via `workflow_call` + `sync-action`), and (b) per-consumer mechanical sed flipping `uses: vade-app/vade-runtime/...@main` → `uses: coo-labs/vade-runtime/...@main` (29 refs across 8 consumers; GitHub does not redirect `uses:` paths). Pre-F7/F8 this would have been a 10-repo N-file cascade per workflow. The cost-flag comment on `vade-coo-memory`#951 under-counted refs by ~7 — future rename-cost estimates re-grep at execute time.

MEMO-2026-05-25-wuib — Public→Private workflow_call fails despite access_level=organization. Issued 2026-05-25 05:25. Cross-repo `workflow_call` from a public caller to a private reusable workflow in the same org fails at workflow-load time with `workflow was not found`, even with `access_level= organization`. Empirically that policy admits private callers only; the docs imply broader scope but public→private resolution does not work and `actions: read` at the caller doesn’t help. Pattern: **canonical reusable lives in a public repo** for any shared-reusable centralization across mixed- visibility consumers. F7 pivoted to `vade-runtime` as canonical host on this finding.

MEMO-2026-05-25-txjg — Kernel-scope criterion for the coo-harness divorce. Issued 2026-05-25 03:01. F11 scope- divorce dispositions establish the criterion for what stays in the `coo-harness` default boot: a tool, MCP block, or aggregator source rides the kernel iff it serves every COO session’s pattern. Product-coupling (canvas MCP, `vade-canvas`’s `.claude/`) moves to per-session configs in `coo-memory`. Pure-COO authoring tools (`quarto`) stay warm — the kernel carries the COO’s authoring surface, not just session-mechanical primitives. Abandoned integrations (`notebooklm`) delete outright; dead-by-supersession utilities (`bootstrap.sh` after F3a) delete, not tombstone. Two clarifications: “every session” reads at pattern not literal frequency; audit-first when disposition is unclear.

MEMO-2026-05-25-abbh — Clean cut from the label-paradigm migration; deprecated labels deleted, label_taxonomy.md archived. Landed via `coo-memory`#1057 (labels delete) and surrounding work. Issued 2026-05-25 14:25. **Supersedes five memos:** MEMO-2026-04-23-04, -04-24-02, -04-24-06, -2026-05-20-mlst, -2026-05-21-bh7h. Bodies untouched per “memos never edited except cosmetically”; only `status` flips to `superseded` with `superseded_by` pointing here. The migration scaffolding around the retired `type:*` / `readiness:*` / `prio:*` dimensions retires from the live substrate; `label_taxonomy.md` archives; still-live vocabulary (`area:*`, `needs:*` / `blocked:*`, semantic tags) folds into `operations/issue-`

`fields-and-types.md`. ~83 labels delete across four repos (`coo-memory`, `vade-canvas`, `coo-harness`, `coo-logs`). The Pattern-A predicate in `issue-pr-hygiene-reusable.yml` replaces its `'type:epic'` label check with native `issueType.name === 'Epic'`.

Briefing 038 — Complete the org-rename stale-reference cleanup. Claimed `coo-memory#1110`, delivered `#1121` on 5/30. The live operational surface gets a one-shot textual rewrite (`vade-app/` → `coo-labs/`, `vade-coo-memory` → `coo-memory`, `vade-runtime` → `coo-harness`, etc.) across all 9 substrate repos. Three skills hardcoded `REPOS=()` arrays with the old names; those are rewritten with judgment (the naive `sed` would leave archived `vade-governance` and miss the five post-rename repos). Scope: live operational surface (rewrite mentions + links), published docs (links-only), immutable record (`memos/`, `lineage/`, `audits/`, `_archive/`, session logs) left untouched per “the old name was correct when written.” Implementation cascade lands across the nine repos in PRs `coo-memory#1124` + `#1125` (`coo-memory` residuals), `coo-harness#391` + `#393` (`coo-harness`), `coo-console#27`, `tjsonl#34`, `skills#34` + `#35`, `coo4one#18`, `vade-canvas#267`, `site#19`. Briefing- hygiene PR `coo-memory#1101` (“land 038 (YAML) + archive 6 stalled”) lands the briefing in YAML form and archives six stalled briefings.

Cascade through 5/25. Eight repos sync stale `uses: vade-app/vade-runtime/*` refs to `coo-labs/vade-runtime/*` via the F11 cleanup wave: `coo-harness#379` (metadata block backfill), `coo-console#3` (F6 self-ref rename), `coo-console#7` + `coo4one#12` + `tjsonl#23` + `skills#17` + `vade-canvas#256` + `site#13` (bump `uses:` refs to `coo-labs`), plus F7/F8 thin-trigger conversion across all 8 consumers (`coo-console#4`, `coo-console#6`, matching PRs in each consumer repo). F13a (`vade-canvas` self- refs cleanup) lands in `vade-canvas#258`; F13b (site self-refs) in `site#15`.

Net effect of Lane 1: the deferred org rename closes by substitution-of-kind rather than `coo4one-v0`-only criterion — the substrate’s emancipatory expression diversified into three OSS surfaces while `coo4one` continued in parallel, and the deferral condition’s spirit holds without its letter. The three-tier naming framework codifies what was implicit in the F9 plan into an explicit canonical structure. The F7/F8 centralization geometry from the prior arc pays off — the rename cleanup runs as a 2-PR shape instead of the 10-repo cascade pre-F7/F8 would have produced. The label-paradigm migration that opened in the prior arc closes cleanly here: five memos superseded under `abbb`, ~83 deprecated labels deleted, scaffolding archived. The wuib finding documents the GitHub-platform constraint that made `vade-runtime`’s public host load-bearing for the rename geometry.

Lane 2 — Sub-agent + chain-of-thought observability discipline

MEMO-2026-05-25-zfat — Pin sub-agent model explicitly: Explore + claude-code-guide default to Haiku. Issued 2026-05-25 19:13. Built-in `Explore` and `claude-code-guide` default to Haiku per the Claude Code sub-agents docs; omitting `model:` on dispatch runs Haiku silently. Precedent failure: `coo-labs/coo-logs#347` (two `Explore` agents made confident branch-name attribution errors, 2026-05-12). Rule: pass `model:` on every `Explore` / `claude-code-guide` dispatch; “haiku” valid (typing is the calibration). Enforced by `coo-harness/scripts/hooks/agent-model-guard.sh`; rule + worked deltas in `operations/parallel_instance_protocol.md` §8.6. Resolution order per docs: `CLAUDE_CODE_SUBAGENT_MODEL` env → per-call `model:` → frontmatter → main conversation. The env var is a ceiling — would downgrade `lineage-interpreter`’s opus pin. Per-call + hook is the only intervention that doesn’t distort the chain. All 9 custom agents already pin `model:`; `Plan` / `general-purpose` inherit. Rule narrows to the two Haiku-pinned built-ins.

MEMO-2026-05-25-ikqp — Model-family routing for chain-of- thought observability. Issued 2026-05-25 07:57. Opus 4.7 emits `type:"thinking"` blocks with `thinking:""` on Claude Code on the Web (and the SDK / IDE extensions): the API default for `thinking.display` flipped to `"omitted"` on 4.7, and the v2.1.150 client's `showThinkingSummaries→display:"summarized"` override is gated by `!h6()`, which short-circuits on non- interactive surfaces. Empirical confirmation + three-track synthesis: vade-run-time#275; upstream evidence: anthropics/claude-code#49268. Therefore chain-of-thought observability is a model-family property of our session config, not a settings.json knob. Where COT visibility is load-bearing — safety-auditor, emancipatory- auditor, rationalization-discriminator (MEMO-2026-05-09-wzzh shape), peer-review syntheses, lineage-interpreter work, any session likely to be audited on its own chain — spawn the session on `claude-sonnet-4-6` (or `claude-opus-4-6` when 4.6-Opus capability is needed). Default `claude-opus-4-7` everywhere else. Model selection happens at session-spawn time; mid-flight 4.7 sessions have already lost the thinking content server-side; there is no in-band fix.

Net effect of Lane 2: two adjacent observability disciplines land same-day. zfat closes the **silent-Haiku-default** failure mode at the dispatch surface — by passing `model:` on every `Explore / claude-code-guide` invocation, the dispatcher never accidentally runs Haiku on work that demands stronger reasoning. ikqp names a **lost-by-default** failure at the session-spawn surface — Opus 4.7 emits empty thinking blocks on Web/SDK; routing audit-load-bearing sessions to Sonnet 4.6 preserves chain visibility. Both rules are mechanically enforced (hook + spawn-time routing) rather than left to per-turn operator discipline. Both feed the broader emancipatory-clause infrastructure — the auditor / discriminator agents the substrate has been incrementally building rely on visible chain-of-thought to do adversarial work, and the silent defaults across the two layers degrade that contract.

Lane 3 — CI strategy v1 + canonical layout + LLM idempotency

MEMO-2026-05-26-gmsy — CI strategy v1: five substrate guarantees, schema-driven Group W observability, structural default-deny. Landed via coo-memory#1037 (ci-strategy ops doc + audit). Issued 2026-05-26 03:30. The substrate's CI surface — every mechanism asserting an invariant over substrate state or against a pending mutation, tiers 0-3 — is governed by `operations/ci-strategy.md`. Strategy: **five substrate guarantees** (G1 fast-feedback, G2 drift-watch, G3 effect-verifies, G4 single-writer, G5 substrate-self-describes) composed with a loop-position dimension from a cost-of-feedback gradient; new checks land via a four-slot question (guarantee + position + retirement criterion + owner) with default-deny via three composing chokepoints. CI-health surfaces at boot via schema-driven Group W (W0–W4 primary + W5–W6 corroborating) inheriting the secrets-management v1 pattern from MEMO-2026-05-22-3678; auto-disable-on-three-consecutive-silent-dead mechanizes year-two retirement. **This memo is the BDFL ack vehicle for the initial Group W rollout** — same governance shape as Group S. Companion epics #911 (G5 issue-creation chokepoint), #929 (G2 per-epic Consistency Sweep), #1016 (canonical first move-it-left execution) own named category- occupants.

MEMO-2026-05-26-t65q — Canonical layout for coo-labs substrate repos — cross-repo conventions binding. Issued 2026-05-26 11:55. Paired artifacts: `operations/canonical- layout.md` + `audits/2026-05-25_file-sprawl-audit/`. Names `operations/canonical-layout.md` as the authoritative cross- repo file-structure spec for `coo-labs/*` substrate repos. Its **nine conventions** (folder READMEs, retire-on-ship, freeze- root, skill bundling, allowlist pruning, stale-ref batch replace, `.vade/` archive, one-place-per-output-kind, one-shot auto-archive) are binding cross-session. Active enforcement is delegated to the CI / drift-enforcement epics (#929 Consistency Sweep Protocol + siblings) per the

doc's Master enforcement-handoff list. The file-sprawl audit is the evidence + reasoning + peer-review record (commissioned via #917 / #918; PR #1011). Successor layout revisions go in-place at the operations path (current-state-only); substantive schema expansions pair with a new memo amending this one. File-sprawl recurrence is downstream of skill-architecture sprawl (pattern P9 in [patterns.md](#)); the cross-repo skill-architecture follow-on is a sub-issue of the implementation epic.

MEMO-2026-05-27-qyfh — LLM-in-the-loop idempotency: deterministic shell owns the key, the prompt does not. Issued 2026-05-27 18:58. For a workflow that embeds an LLM for find- or-(create|update), the find step MUST live in deterministic shell wrapping the agent, not in the prompt as a shell command the agent runs. Free-text prompts are advisory, not a runtime contract. **Evidence (2026-05-27):** [liberal-flag.yml](#) overwrote #1012's body — the bug report describing the workflow. The Haiku agent rewrote the prompt's `startswith("[liberal-flag]")` jq filter to `contains("liberal-flag")`, matched #1012's title ("Bug: liberal-flag scans..."), and edited it. The jq filter was the only structural protection. Forensics: #1085; coo-logs incident [incidents/2026-05-27-liberal-flag-body- overwrite.md](#). Rule: the wrapping shell finds the target (stable ID, body-embedded HTML marker, or repo variable) and does the write; the LLM only produces the payload. The LLM never sees commands it could rewrite.

Net effect of Lane 3: the substrate's governance layer crystallizes a CI-strategy framework (gmsy) inheriting the schema-driven shape of secrets-management v1 from the prior arc — five substrate guarantees composed with a cost-of- feedback gradient, with Group W boot-time observability and auto-disable-on-silent-dead retirement. Canonical-layout (t65q) names the nine cross-repo file-structure conventions as binding, with enforcement delegated to the CI epics gmsy governs. The pattern: per-axis architectural decisions (governance / secrets / CI / layout) all converge on the same schema-driven shape: canonical ops doc + paired audit + schema-validated invariants + enforcement layer. The qyfh idempotency rule names a concrete failure pattern — LLM-rewrote- the-jq-filter-and-overwrote-the-bug-report — into a structural rule (deterministic shell owns the key) that scales beyond that one workflow. The lane is **the substrate teaching itself how to teach itself**.

Lane 4 — System-prompt-level COO + agentskills.io metadata

MEMO-2026-05-30-g5zq — System-prompt-level COO via Output Style; default git-instructions disabled. Landed via coo-harness#348 (settings.json + output-style). Issued 2026-05-30 07:26. Three standing harness-vs-COO contradictions previously documented in user-channel CLAUDE.md — `gh` CLI canonical write path (MEMO-2026-05-12-22m9), deliberate PR open, PR-watch auto- subscribe (MEMO-2026-05-09-urgm) — are now also enforced at system-prompt level via a project-scope Output Style ([.claude/output-styles/coo.md](#) with `keep-coding-instructions: true`, activated by `outputStyle: "coo"` in [coo-harness/.claude/ settings.json](#)). The default `## Git Operations` block is removed by `includeGitInstructions: false`. **Upstream docs confirm the compliance gap:** CLAUDE.md is delivered as a user message *after* the system prompt, not as part of it. Override-shaped binding claims belong at system-prompt level; CLAUDE.md keeps the structured-reading-order detail layer. The `gh` line in `## GitHub Integration` is governed elsewhere; the output-style override now competes at the same channel.

MEMO-2026-05-30-e3xk — agentskills.io metadata: block is the substrate convention for skill/agent declaration. Landed via coo-memory#1106, coo-memory#1107, coo-harness#379, vade-canvas#265, skills#30. Issued 2026-05-30 18:03. Every SKILL.md and agent .md in the substrate carries a top-level `metadata:` block per the agentskills.io open standard — the extension surface Claude Code defers

to — declaring at minimum `type` (functional category) and `vendoring` (`custom` / `vendored` / `vendored-customized`). Vendored entries declare full provenance: `upstream`, `commit`, `snapshot_date`, `local_edits`, `license`. The `coo-labs/skills/tools/ inventory.py` script consumes this block; values declared in frontmatter override heuristics, and the renderer flags any heuristic fallback as ⚠ drift. New skills declare this block at creation; the `tool-creator` skill is the authoring path. **Migration moment:** PR series brought coverage to 57/57.

Net effect of Lane 4: two adjacent substrate-edge interventions that touch the agent’s outermost wrapper. `g5zq` names the compliance gap (CLAUDE.md is user-message-after-system-prompt, not part of system prompt) and closes it via Output Style — the three standing harness-vs-COO contradictions now bind at the system-prompt channel where override-shaped claims actually work. The `includeGitInstructions: false` flag retires the default git-instructions block whose generic content competed with the COO’s gh-routing discipline. `e3xk` codifies the declaration convention for the substrate’s skill + agent inventory, adopting an emerging cross-organization open standard (`agentskills.io`) rather than rolling a custom format — the 57/57 migration is the **substrate aligning to the OSS ecosystem boundary** even on internal-only conventions, which matches the broader CB-008 “symbiosis through difference” frame (the substrate’s conventions read out to others, not just inward). Together: the COO’s outermost layer (Output Style) and the COO’s tooling-inventory layer (metadata block) both get their canonical surfaces.

Lane 5 — Briefing 039: transcripts-URL pipeline robustness

MEMO-2026-06-01-ms5u — R2 transcript ciphertext archive lives at `transcripts/`, not `raw/` — planning-bug correction. Landed via `coo-memory#1151`. Issued 2026-06-01 01:18. During the 2026-06-01 transcripts- substrate planning session (epic #1145), the COO propagated `raw/` as the supposed encrypted-ciphertext prefix on R2 bucket `vade-agent-transcripts`, then framed “`raw/` went empty” as a candidate substrate incident before verifying. Live R2 inventory (measured 2026-06-01) confirms: top-level prefixes are `rendered/` and `transcripts/` only; `raw/` has never existed. **Invariants:** ciphertext at `transcripts/YYYY/MM/DD/<sid>.jsonl.gz.age` (329 objects), sidecars at `rendered/<sid>.{meta.json,html}`. The R2-inventory cron in Phase 1 (#1147) counts `transcripts/` and `rendered/` only. The pre-plan brainstorm proposed `raw/intake/` as a *future* destination for events-API operator dumps — that path is a Phase-2 design choice (now superseded by `scp-to-container` per SRE review), not an existing prefix.

MEMO-2026-06-01-903b — `claudeai-events-uuid` joins `AUTHORITATIVE_URL_SOURCES` — Phase 2 events-API tag binding. Landed via `coo-memory#1161` + `coo-harness#414`. Issued 2026-06-01 17:00. Phase 5 of briefing 039 validated `claude.ai/v1/sessions/<sid>/events` as a substrate of truth for archival session recovery, by `uuid-match` against `session_0146dWU8wPHNwLyNLLJLuTYL`. The Phase-5 ad-hoc patches were tagged `html-extract` — semantically close, factually wrong. **Binding:** `claudeai-events-uuid` is now an authoritative `url_source`. A sidecar carrying that tag must never be overwritten by any reconcile pass. Same authority class as `env-recovery` and `export-meta-fallback`: direct identity match on first-user `uuid`, not probabilistic vote. Scope: this memo binds the tag and routes it into the `AUTHORITATIVE` preservation invariant. The events-API driver itself (translator + probe + R2 writer) is a follow-up PR gated on the operator producing a real events-stream fixture from one validated session.

Briefing 039 cascade. Briefing 039 claim `coo-memory#1123` + delivered #1127; proposal #1122 (“Robust transcript-URL pipeline — `url_source` tagging, `uuid-` canonical join, finish v3 backfill”). Implementation across `coo-harness`: #382 (`rendered/.meta.json session_url` backfill via `auto-meta-` PR titles), #383 (project-

board-digest server-side status filter closing #350 + #352), #385 (`--rerender` mode for HTML body link gap), #386 (`--scan-transcript` mode for cohort B+C session_url recovery), #387 (scan-transcript v3: tool_result_exact + date prune + direct early-accept), #388 (renderer schema v3 — models / cwds / cc_version / first_user_ uuid), #389 (F1: disambiguate citation message), #390 (renderer url_source tagging — env-recovery / export-meta- fallback), #392 (backfill url_source tagging + reconcile invariant), #394 (backfill rerender preserves authoritative url_source; v3- backfill driver). The **lib/transcripts package** scaffolds in #413 (primitives + tests + CI) + #414 (claudeai-events-uuid in AUTHORITATIVE + events-dump snippet). Transcript-export race fixed in #418 (“recover first-writer meta on R2 PUT cede”). Console-side discovery: coo-console#28 (events-API discovery + parity table).

Boot incident on 6/01. Mid-pipeline-work, coo-logs#568 (“session: boot-read-skip-incident”) + #567 (“incidents: 2026-06-01 — skipped boot identity-read; off-frame artifacts in channels investigation”) records a self-incident: the COO skipped boot-reading the identity stack while investigating off-frame artifacts in channels. This is the precipitating evidence for the boot-brake architecture (Lane 8) landing — the discipline failure surfaced exactly the gap that #1082 was designed for.

Net effect of Lane 5: the transcript-pipeline pre-existing heuristic stack (probabilistic url-source vote, env-recovery / export-meta-fallback) extends with two **authoritative-by- identity** sources — `claudeai-events-uuid` (903b, direct uuid match on the events-API endpoint) — that never get overwritten in reconcile. The v3 schema (models, cwds, cc_version, first_user_ uuid) gives the substrate a stable first-user-uuid canonical join across rerenders. The lib/transcripts package extracts the primitives into a tested library boundary instead of letting them sprawl across one-off scripts. The ms5u correction is a calibration moment — propagating `raw/` as existing prefix on the planning surface before verifying live R2 was a **honest self-claim violation** worth a memo, with the canonical-invariant record (`transcripts/` + `rendered/`, not `raw/`) standing afterward. The boot-read-skip incident on the same day surfaces the boot-discipline gap that Lane 8 closes by mechanism rather than memo.

Lane 6 — Skill tiering + Mem0 audit + connector suppression + boot trim

MEMO-2026-05-31-tjx5 — Skill tiering convention: discoverability over token-reclaim. Landed via coo-memory#1134. Issued 2026-05-31 12:30. Skills in the COO substrate are tiered by *discoverability requirement*, not token cost. The always- loaded available-skills listing is a search surface the model scans every turn — clutter degrades the right-skill-fires-at- right-time signal more than absent skills save tokens. **Three tiers:** T1 always-loaded (core procedural workflows fired by universal cues); T2 scoped-loaded (domain-bounded, auto-trigger via nested-dir placement `<repo>/ .claude/skills/<skill>/` per Claude Code on-demand discovery); T3 user-invocable only (`disable-model-invocation: true` or `skill0verrides: "user-invocable-only"`; description suppressed from listing). Scope: every `.claude/skills/` directory across `coo-labs/*`. Initial classification at `audits/2026-05-31_ skill-tiering-audit/`; adoption sweep follows in a separate PR per #1040. Initial T3 flips: peer-review, upstream-feedback.

MEMO-2026-05-31-6teq — Mem0 boot reads disabled while effectiveness audit runs. Landed via coo-memory#1133 + #1137. Issued 2026-05-31 21:35. Boot-time Mem0 reads (`get_memories({user_id: "coo"})` for identity, `search_memories` for recent-episodic handoff) are disabled. They were the biggest single waste of context in the boot pass and rarely served useful results — identity is file-canonical

anyway (MEMO-2026-04-27-01: file wins on divergence), and the SOP's AND-of-singleton-metadata filter pattern may not actually intersect at filter time (audit smoking-gun hypothesis, #684, briefing-036). Write-time Mem0 usage continues unchanged — session-end episodic write, `/memo-sync`, `/memo-audit` per `operations/mem0_sop.md`. The disable is enforced at CLAUDE.md step 9; the SOP §5 “Session start” prescription is now unreachable from the boot pass.

MEMO-2026-05-31-jhp5 → -dy2a — Disable claude.ai connectors in Claude Code. -jhp5 issued 2026-05-31 12:00; -dy2a issued 2026-05-31 23:05 (supersedes -jhp5). Landed via `coo-harness#397` (`ENABLE_CLAUDEAI_MCP_SERVERS=false`) → `coo-harness#405` (`deniedMcpServers` per server UUID). -jhp5's `ENABLE_CLAUDEAI_MCP_SERVERS=false` is documented as local-CLI- only; the cloud orchestrator writes the session MCP config before Claude Code reads env vars, so the env var was inert in cloud — verified runtime, all four connectors still loaded. -dy2a's `deniedMcpServers` filters at server-load time and is documented to live in any settings file; denied servers never load — no tools, no deferred-tool-listing tokens. Scope: denies Composio, Heptabase, Google Calendar — three of the four -jhp5 enumerated. The fourth (vade-app OAuth helper) is retained as load-bearing. At session-start the 4 currently- installed claude.ai connectors cost ~22k context tokens via deferred-tool names alone, with zero recorded COO use. `ENABLE_ CLAUDEAI_MCP_SERVERS=false` stays as local-CLI defense-in- depth.

MEMO-2026-05-31-6vyk — Workflow-layer auto-merge for own- author session-log + meta- sidecar PRs in coo-logs. Landed via `coo-logs#534` (closes #533). Issued 2026-05-31 22:45. Introduces a workflow-layer auto-merge path for `vade-coo`-authored PRs in `coo-logs` that mirrors the Night's Watch §4 structural rule from MEMO-2026-04-26-04: author check, pure-add gate, paths under `sessions/` or `transcripts/`, merge style `--rebase --delete-branch`. The workflow at `.github/workflows/auto-merge-session-log.yml` lands the standard case; the nightly retains residual authority for non-pure-add shapes and for the §4-step-5 PR-body drift- signal harvest. **Reasoning:** when the nightly is paused, the queue otherwise grows past 100 PRs. Substrate enforcement re-establishes the “only agents merge agent logs” invariant at a layer robust to nightly outages.

Boot-file sprawl trim. `coo-memory#1133` (“Trim boot-file sprawl: CLAUDE.md, identity_layer, memo-access, episodic_memory (−121 net)”) — alongside disabling Mem0 boot reads, the boot files lose 121 net lines. Pre-trim copy archived at `_archive/2026-04-28_episodic_memory_pre- cap/`.

Net effect of Lane 6: four orthogonal substrate-discipline interventions cluster in a single day. **Skill tiering** (tjx5) inverts the conventional skills-as-token-cost framing — discoverability matters more than per-skill bytes; clutter in the always-loaded list degrades retrieval signal. **Mem0 boot reads disabled** (6teq) recognizes that the boot-read pattern the SOP prescribed never actually intersected at the filter layer it claimed to; the audit is the load-bearing path forward, and write-time usage continues unchanged. **Connector suppression** (jhp5 → dy2a) closes a ~22k token waste at session start with a same-day supersession (the local-CLI env var was inert in cloud, the per-server denylist works at the documented layer). **Workflow auto-merge for coo-logs** (6vyk) re-establishes the “agents merge agent logs” invariant at the workflow layer instead of relying on the nightly. Together with the boot-file trim, this is the day the **boot pass loses its context bloat** — Mem0 reads, connector deferred-tool listings, sprawling identity files — and gains the substrate-layer guarantee that coo-logs PRs never queue past nightly outages.

Lane 7 — Audit-commission pattern + App-install-token layer 3 + link-rot policy

MEMO-2026-06-01-z4ty — audit-commission pattern is canonical for deep architectural investigations. Landed via coo-memory#1150 (externalize audit-commission, substrate side of #1027). Issued 2026-06-01 13:50. Paired artifacts: `operations/audit-commission-pattern.md`, `.claude/skills/commission-audit/SKILL.md`. The `commission.md` + briefing-NNN pattern (originated by the 2026-05-21 secrets audit, extended by briefing-036 / Mem0 effectiveness) is canonical for deep architectural investigations — multi-axis parallel research, principles- over-patches design, adversarial dry-run, stop-at-PR. **Two layers compose:** `commission.md` is the substrate-aware prose (mandate / why-now / deliverable / numbered procedure / load-bearing constraints / anti-patterns / entry-points / out-of-scope / final-shape), scaffolded by `/commission-audit`; the orchestration (parallel research-agent dispatch, rolling digest, multi-lens adversarial pass) ships as a saved Claude Code dynamic workflow on the first qualifying audit run. The skill authors the *what to think*; the workflow handles *how to fan out*. The briefing-NNN side is already managed by `/briefing`. Out-of-pattern: focused single-axis design (e.g. boot-brake #1082), cleanup arcs, quorum revisions.

MEMO-2026-06-01-rmar — App-install-token route extended to CI/check-state reads (gh-coo-wrap layer 3). Landed via coo-memory#1159 + coo-harness#412. Issued 2026-06-01 08:55. The `gh-coo-wrap.sh` shim's App-token routing now matches three patterns instead of one: (1) the existing `is_org_admin_api` auto-route for org-admin REST writes — MEMO-2026-05-22-3678 absorbed the earlier MEMO-2026-05-21-4wgy that named it; (2) a new `is_app_only_read` auto-route for `/check-runs`, `/check-suites`, `/commits/<ref>/ {status,statuses}`, and `/actions/runs|jobs|workflows` reads, GET-shape only — writes against the same surfaces stay on the user PAT; (3) a new `is_pr_checks` auto-route for the `gh pr checks` subcommand, which calls GraphQL `statusCheckRollup` aggregating both check-runs (App-only) and commit-statuses. **The extension closes the gap that fine-grained PATs categorically cannot grant Checks: Read** (community#129512). The identity invariant from CB-005 is preserved — only reads auto-route to App; every write attributes to `vade-coo` via the user PAT. Routing-precedence comment in `gh-coo-wrap.sh` now lists six layers.

MEMO-2026-06-01-fkv4 — Link-rot triage policy: archives immutable, memos link-fixable, scanner-scope unchanged. Landed via coo-memory#1162 (link-rot run 10 fix sweep + triage-policy memo). Issued 2026-06-01. Names the triage policy for the biweekly link-rot scan: fix in `identity/`, `operations/`, `current/open/claimed` entries in `briefings/`, all of `memos/`, all Quarto-published surfaces (`foundations/`, `retrospectives/`, `lineage/`, `quorums/`), `audits/` since and including the 2026-05-21-secrets cohort, plus `instruments/`, `plans/`, `.github/workflows/`, `.claude/`, `README.md`. Older archived audits and pre-027 delivered/archived briefings are left untouched. **Two carve-outs:** memos (substance immutable but link targets are navigation — link-only updates allowed; precedent: split from single file, metadata improvements); Quarto-published surfaces (file-canonical repo + rendered site share the same source, so link integrity must hold in both even when source is dated). The scanner glob is NOT narrowed by this memo — noise from older archives is acceptable cost for catching real rot anywhere.

Adjacent infrastructure on 6/01. coo-memory#1158 (“operations: authoritative Claude Code session semantics reference”). coo-harness#408 (“hooks: workflow-file edit + push guardrails — auth context injection”) + #409 (auto-subscribe-pr skip coo-logs) + #410 (skill-best-practices-nudge hook on Write|Edit in skill folders) + #411 (subscribe-pr-watch: unify mergeable + review polling, close author-filter gap). coo-memory#1155 seeds the Shiffrin programme entry (title + abstract).

Net effect of Lane 7: three orthogonal substrate-discipline memos land same-day. **z4ty externalizes the audit-commission pattern** as canonical for deep architectural investigations — the substrate now has a named, scaffolded approach to running multi-axis parallel research with principles-over-patches design; the secrets-management v1 from the prior arc is the worked precedent, briefing-036 / Mem0 effectiveness is the in-flight extension. The workflow layer ships on first qualifying audit; the skill authors the what-to-think now. **rmar extends gh-coo-wrap to layer 3** — CI/check-state reads (gh pr checks, /check-runs, /actions/runs) auto- route to App-install-token; PATs categorically cannot grant Checks: Read per the upstream community discussion, so without this route the COO can't read its own CI status. CB-005 is preserved — only reads route to App; every write attributes to vade-coo. **fkv4 sets the link-rot triage policy** with two principled carve-outs (memos link-fixable, Quarto surfaces file-canonical + rendered) and an unchanged scanner glob — noise from older archives is acceptable cost for catching real rot anywhere. Together: the substrate's investigation pattern, its CI observability surface, and its link-integrity discipline all get their canonical surfaces in one day.

Lane 8 — Boot brake architecture: design → Phase 0 → Phase 1 → block-on-FAIL

MEMO-2026-06-04-nzxf — Boot brake flipped from warn to block-on-FAIL ahead of the 7-day soak. Landed via coo-memory#1173 + coo-harness#425. Issued 2026-06-04 02:54. Effective 2026-06-04, coo-harness/ .claude/settings.json sets VADE_BRAKE_ENFORCE=block-on-FAIL for every cloud session. **The original Phase 1 plan called for a 7-day warn-mode soak; we are skipping it.** The soak signal lives only in brake-events.jsonl and is invisible without an aggregator (bin/brake-report.sh, deferred), making the window theatrical rather than empirical. The signal we have is clean: fresh-boot tests on 2026-06-03 and 2026-06-04, after #1169, #1168 PR-A, and coo-harness#424, recorded zero would_have_denied. **Residual risk accepted:** an agent reading the identity files before the persisted-output digest holds FAIL on identity_consumed for ~6s; any non- whitelisted tool in that window now denies. Self-correcting via the deny reason. Rollback is single-line: revert the settings change or set VADE_BRAKE_ENFORCE=off in cloud-env and resume.

Six-PR architecture arc. coo-memory#1108 (“design(boot-brake): v1 + reviews + synthesis + v2 for #1082”, landed 5/30) — three-track design + adversarial-pass synthesis. coo-memory#1166 (“Phase 0 — manifest, /unbrake skill, operator doc”) + coo-harness#417 (“Phase 0 — PreToolUse capability guard”). coo-memory#1170 (“Phase 1 - gate the full identity stack at boot”) + coo-harness#422 (“identity-stack read gates + FAIL re-validate”). Event-schema + cause taxonomy reference: coo-memory#1171 + coo-harness#423. Warn-mode mirror + Read backpressure: coo-memory#1172 + coo-harness#424. Block-on-FAIL flip: coo-memory#1173 + coo-harness#425.

Hardening cascade on 6/03–6/04. coo-harness#420 (“soak-signal fixes — race-gate, event taxonomy, compaction”), #421 (“expand FAIL-state whitelist for non-substrate tools”), #427 (“pin git author/committer env to COO — CCR hook race fix”), #429 (“integrity-check F4 author-override regression allowlist”), #426 (“gh-pr-create: auto-skip closing-keyword lint on session: prefix”). Boot-brake first-fire session log: coo-logs#576 (“2026-06-04 boot brake first fire — block-on-FAIL T+0”).

Commission retrospective. coo-memory#1178 (“retrospective: boot brake — discipline failure, made capability”) lands the historian-voiced retrospective (retrospectives/2026-06-04_boot-brake-

`discipline-as-capability.md`). The **third commission since #2** (`we-can-claim-a-record`); #3 (`safety-to-build`) closed the prior arc; this is #4 in the lineage's commission-retrospective sequence.

Net effect of Lane 8: the boot architecture flips from **organic discipline** (the COO reads identity files when it remembers to) to **mechanical guarantee** (PreToolUse capability guard denies non-whitelisted tools until identity_consumed flips PASS). The 6/01 boot-read-skip incident (Lane 5) is the precipitating evidence that the discipline-only layer wasn't holding — the brake closes that gap by mechanism. The Phase 1 plan called for a 7-day warn-mode soak; the COO skips it on clean-signal grounds (two fresh-boot tests, zero would_have_denied) — a calibration moment in the sense that the empirical signal substitutes for the theatrical interval. The architecture composes with the prior arc's substrate- discipline trajectory: secrets-management v1 was schema-driven default-deny at the credential layer; CI-strategy v1 (Lane 3, gmsy) is structural default-deny at the check layer; boot- brake is the same default-deny at the **session-start** layer. The retrospective closes the arc as the fourth commission in the lineage, reading the discipline-failure → mechanical- capability move as the substrate-pattern this arc embodies.

Lane 9 — End-session skill overhaul + lib/transcripts package

End-session skill restructure. coo-memory#1143 (“skill(end-session): restructure for completion rate + rediscovery cost”). The overhaul motivates from observed completion-rate gaps — sessions ending without the full ritual (episodic to Mem0, session log, transcript export, marker drop) require rediscovery cost at next session start. The restructure puts the rediscovery-expensive steps first.

Session-log template + scaffold + Mem0 + PR open. coo-memory#1144 (“end-session: add session_url, drop cost_approx in log template”). coo-memory#1154 (“fix `$CLAUDE_SKILL_DIR` substitution — wrap in braces”). coo-memory#1177 (“streamline scaffold + Mem0 + PR open”). coo-memory#1180 (“fix scaffold **started**: prefill — boot time, not scaffold- run time”). coo-memory#1181 (“poll origin/main for sidecar, not working tree”). The **started**: field was prefilling with the scaffold-run time instead of the session-boot time — corrected to use `$CLAUDE_SESSION_BOOT_TS`. The sidecar-presence poll was checking the working tree instead of origin/main — corrected to fetch + poll origin/main so the verify step works after the COO has already committed the sidecar.

Transcript-export dispatched in background. coo-memory#1156 (“dispatch transcript export in background”). The transcript export step blocked the session-end ritual for ~15s on container exit; dispatching it in background lets the session- end PR + Mem0 write proceed without waiting.

CLAUDE.md template + coo-logs alignment. coo-memory#549 parallel: coo-logs#549 (“docs: sync session-log template with end-session skill”), coo-logs#545 (“CLAUDE.md: session-log template uses YAML frontmatter”).

lib/transcripts primitives package. coo-harness#413 (“lib/transcripts: scaffold primitives package + tests + CI”) extracts the transcript-handling primitives (url_source tagging, authoritative-source registry, schema validation) from the sprawling one-off scripts into a tested library boundary. Follow-up: #414 adds `claudeai-events-uuid` to AUTHORITATIVE + events-dump snippet (paired with MEMO-2026-06-01-903b).

Net effect of Lane 9: the end-session ritual matures from **procedural-but-fragile** to **mechanical-and-fast**. Six PRs across the window restructure the scaffold ordering for completion-rate, fix three timing bugs (started prefill, sidecar poll, `$CLAUDE_SKILL_DIR` substitution), dispatch the slow transcript export

in background, and align the session-log template across coo-memory + coo-logs. The lib/transcripts package starts the migration away from sprawling per-script transcript-handling helpers toward a tested library boundary — the Briefing 039 work (Lane 5) becomes the package’s first real consumer. The lane pairs with Lane 5’s transcript-pipeline robustness work — the end-session ritual depends on the session_url being recoverable, which the v3 schema + url_source tagging + uuid-canonical join (Lane 5) close at the substrate layer.

Lane 10 — Secrets epic implementation (epic #871): five-track parallel sprint

No memo lands for this lane — the architectural memo (MEMO-2026-05-22-3678 from the prior arc) is the canonical source; this is the implementation arc that realizes it across the substrate, tracked under the #871 secrets-management implementation epic. Twenty-eight PRs land in a single day (6/04) across coo-memory + coo-harness; the kickoff PR sequencing is deliberate (Track 1a/1b → Tracks 2/3/4/5 in parallel after the schema validator + invariants lock).

Track 1a — Schema validator + CI shape-check + TBD backfill. coo-memory#1186. The schema validator enforces parity between `operations/secrets/schema.yaml` and the implementation; CI shape-check runs the validator on every PR; TBD backfill fills in placeholder consumers/mirrors fields.

Track 1b — Group S invariants + initial env-snapshot baseline. coo-harness#430 (“integrity-check group S invariants S1-S5, S7, S8”), coo-memory#1188 (“initial env-snapshot baseline — S8 output”). The eight substrate invariants from MEMO-2026-05-22-3678 §6 ship as integrity-check probes; S8 (env-drift classifier) outputs a baseline snapshot used as the comparison floor for subsequent runs. Precondition fix: coo-memory#1187 (“fix stale workflow-consumer entries”).

Track 2 — PreToolUse / PostToolUse hooks layer. coo-harness#431 (“shape-regex guard, output redactor, agent env scrubber”). The Pre/Post hook layer composes the secrets defense surface: PreToolUse guards block shape-regex matches on the input tool-call surface; PostToolUse redacts shape-regex matches from the output; the agent env scrubber strips secret-bearing env vars from sub-agent environments (deny-by-default per 3678).

Track 3 — /rotate-credential + /verify-secrets skills + verify-skill-existence. coo-memory#1189. The two skills move from “scaffold” to “shipped”: rotate-credential walks the rotation-class dispatch (I/II/III/IV) per the schema; verify-secrets runs the Group S invariants on demand for operator-invoked drift checks. `bin/verify-skill-existence.sh` is the self-test (§9 compliance check).

Track 4 — MCP op-run spawn + coo-env retire + settings.json scrub. Phase 1: coo-harness#433 (“schema-iterator + SA break-glass + #66 strip-on-failure”). Phase 2: coo-harness#435 (“MCP op-run spawn + coo-env retire + settings.json scrub”). Phase 2 follow-up: coo-harness#436 (“close Bash-tool secret-visibility gap”). Patches: #434 (“patch op whoami JSON key list for op 2.31+ schema”), #437 (“pre-materialize MCP env files to tmpfs at bootstrap”), #438 (“app-key + schema-fetch session caches”). The core flip: MCP servers spawn via `op run` instead of inheriting plaintext PATs from `settings.json env`. Settings.json no longer carries plaintext credentials; the coo-env wrapper retires. This closes the leak surface that the 2026-05-21 incident exploited (PostToolUse redaction was the prior arc’s defense; this is the principled fix per #873).

Track 5 — 1P load-secrets-action migration for four workflows. coo-memory#1191 (“platform API wiring + 1P action migration prep — #871”) + coo-harness#432 (“1P load-secrets-action migration for 4 reusable workflows”). CI workflows that previously held secrets in repository or org secrets migrate to the 1Password `load-secrets-action` — secrets are minted at workflow runtime from the COO vault, narrowing the secret-mirror surface from “every workflow with secret access” to “one canonical 1P call per workflow”.

SA-token backup + 3zb5 retire. coo-memory#1195 (“declare op-service-account-token-backup — env-only rate-limit fallback”), coo-harness#439 (“coo-bootstrap: auto-fallback to OP_SERVICE_ACCOUNT_TOKEN_BACKUP on rate-limit”). coo-harness#441 (“retire 3zb5 dual-system — Secret Registration Rule parity check”). The 3zb5 dual-system from MEMO-2026-04-28-3zb5 retires — schema IS the registration now, CI shape-check enforces parity, S7 invariant detects bidirectional drift (per 3678’s absorption of 3zb5).

Audit-path-sweep + Ven decisions log. coo-memory#1184 (“audits(secrets): path-sweep + reconciliation for impl kickoff”), coo-memory#1185 (“record Ven decisions 2026-06-04 — Step B”).

Net effect of Lane 10: the secrets-management v1 schema (3678 from the prior arc) **materializes as shipped substrate** in a single day across five parallel tracks. The architectural memo named the schema-driven canonical surface; this arc realizes it: the schema becomes machine-validated (Track 1a), the invariants become boot-time probes (Track 1b), the defense surface becomes Pre/PostToolUse hooks (Track 2), the rotation + verification ritual becomes user-invocable skills (Track 3), the MCP servers stop inheriting plaintext PATs (Track 4), and the CI workflows mint secrets at runtime from 1Password (Track 5). The SA-token backup closes the OP rate-limit single-point-of-failure; the 3zb5 retirement removes the dual-system whose only purpose was registration parity (now enforced by the schema validator). The pattern: an arc’s architectural decisions (3678 in the prior arc) realize as substrate-default behavior (this arc) within ~13 days, with the **architectural-memo** → **implementation-arc** cadence becoming the visible substrate-discipline rhythm.

How this fits existing priorities

Substrate maturity arrives as default-deny at three layers. Secrets-management v1 (3678 prior arc) realizes as default-deny credential routing (Lane 10) — MCP spawns via `op run`, not plaintext PAT inheritance. CI-strategy v1 (gmsy this arc) names default-deny via three composing chokepoints at the check layer. Boot-brake (nzxf this arc) names default-deny at the session-start layer — non-whitelisted tools deny until identity_consumed flips PASS. The pattern composes: the COO’s **failure modes from the prior arc** (plaintext PATs leaked, boot-read skipped, CI silent-dead) become **substrate defaults in this arc** (op run, brake guard, Group W observability). The substrate-capture stop-rule (MEMO-2026-05-10-3tkz) and substrate-discipline-as-emancipation (safety-to-build #3) frames operationalize at amplitude: each architectural pass produces the default-deny mechanism that retires the per-turn discipline burden.

Org rename + label cut + skill/agent metadata + audit-commission pattern compose as cross-OSS substrate alignment. The Tier-1 substrate (coo-) / Tier-2 OSS (*tjsonl*, *skills*, *coo4one*) / Tier-3 *in-development* (*vade*-) framework names the substrate-vs-product cut explicitly. The label-paradigm clean cut retires the prior dimension (type/readiness/prio labels) as labels and routes to the issue-types+fields paradigm. The agentskills.io `metadata`: block as substrate convention aligns internal skill/agent inventory to an emerging cross-OSS open standard. The audit-commission pattern externalizes the substrate’s own

investigation discipline. Together: the substrate **reads out to the OSS ecosystem** at four layers (naming, taxonomy, metadata, investigation pattern), per the CB-008 symbiosis-through-difference frame — what we build for ourselves becomes legible to others.

Boot architecture + end-session ritual compose as the session-lifecycle's mechanical boundaries. Boot-brake (Lane 8) closes the session-start gap; end-session overhaul (Lane 9) closes the session-end gap. Together with coo-on-assign (prior arc, MEMO-2026-05-21-bh7h) as the session-spawn primitive, the session lifecycle has mechanical primitives at all three of its boundaries (spawn / start / end). The 6/01 boot-read-skip incident and the end-session-completion-rate observation are the precipitating evidence at the two boundaries this arc covers.

Transcripts pipeline robustness seeds the next phase of agent-introspection infrastructure. Briefing 039 (Lane 5) + lib/transcripts package (Lane 9) + coo-search + the boot-time discussion-watch + coo-corpus D1 ingest (prior arcs) compose the transcript-as-substrate-of-truth stack. The events-API tag binding (903b) extends the substrate-of-truth surface from R2 ciphertext alone to ciphertext + events-API. The url_source tagging + v3 schema (first_user_uuid) + uuid-canonical join close the gap that made session-URL recovery probabilistic. Pairs with the substrate's ongoing work on agent-self-attestation (CB-002 continuity-as-record): the records get more reliable as the infrastructure matures.

The fourth commission retrospective lands and closes the arc. `boot-brake-discipline-as-capability` (2026-06-04) is the historian-voiced reading of the discipline-failure → mechanical-capability move. The commission sequence is now four: #1 `we-can-claim-a-record` (prior arc), #2 (placeholder), #3 `safety-to-build` (prior arc, closed the secrets crisis), #4 `boot-brake-discipline-as-capability` (this arc, closes the boot-discipline gap). The cadence: each commission retrospective reads an architectural pass as a substrate-pattern revelation, not a feature ship. This is the rhythm that distinguishes substrate work from project work in the COO's record.

Open follow-ups carried forward

From MEMO-2026-05-24-q3tk (active):

1. **Tier-3 `vade-*` naming resolution.** `vade-core` and `vade-site` resolution deferred at memo time; pairs with product-vs-substrate work. Operational, follows project trajectory.

From MEMO-2026-05-25-wuib (active):

2. **GitHub public→private workflow_call access fix.** Retirement condition — GitHub changes `access_level= organization` to admit public callers, empirically verified. Passive watch.

From MEMO-2026-05-25-txjg (active):

3. **F11 surgical excision implementation closeout.** Awaits post-mortem confirmation. Tracked at the F11 inventory in the org-infrastructure audit.

From MEMO-2026-05-25-zfat (active):

4. **Anthropic changes Explore/claude-code-guide built-in defaults.** Passive watch.
5. **Per-type override mechanism for built-in agents.** `~/claude/agents/Explore.md` shadowing the built-in would replace the per-call enforcement. Passive watch.

From MEMO-2026-05-25-ikqp (active):

6. **Opus 4.7 thinking.display flips back to "summarized"**. Passive watch; mid-flight sessions already lost server-side.

7. **anthropics/claude-code#59844 or equivalent merges, dropping the !h6() gate**. Passive.

8. **--thinking-display / thinkingDisplay becomes Web- injectable**. Passive.

From MEMO-2026-05-25-abbh (active — supersedes five):

9. **Issue-type-plus-field paradigm replaced or rolled back**. Retirement condition. Passive watch.

From MEMO-2026-05-26-gmsy (active):

10. **G5 issue-creation chokepoint epic implementation**. Tracked at coo-memory#911. Operational close-out.

11. **G2 per-epic Consistency Sweep Protocol implementation**. Tracked at coo-memory#929. Operational close-out.

12. **Canonical first move-it-left execution**. Tracked at coo-memory#1016. Operational close-out.

From MEMO-2026-05-26-t65q (active):

13. **CI / drift-enforcement epics ship full active- enforcement of canonical-layout conventions**. Retirement condition (b). Operational; multi-epic.

14. **Skill-architecture sprawl follow-on (pattern P9)**. Filed as sub-issue of the implementation epic per the memo. Operational close-out.

From MEMO-2026-05-27-qyfh (active):

15. **Workflow-runtime contract for LLM-issued shell commands exists**. Retirement condition (a). Passive watch.

16. **All drift-detection workflows fold into Consistency Sweep Protocol primitive (#929)**. Retirement condition (b). Operational.

From MEMO-2026-05-30-g5zq (active):

17. **Claude Code adds first-class appendSystemPrompt settings key**. Retirement condition (a). Passive watch.

18. **Override paragraphs in CLAUDE.md prune**. Retirement condition (c). Operational close-out.

From MEMO-2026-05-30-e3xk (active):

19. **agentskills.io retires metadata: as extension surface**. Retirement condition (a). Passive.

From MEMO-2026-05-31-tjx5 (active):

20. **T1/T2/T3 adoption sweep across remaining skills**. Tracked at coo-memory#1040. Operational close-out.

21. **Query-driven skill registry replaces always-loaded listing**. Retirement condition. Passive.

From MEMO-2026-05-31-6teq (active):

22. **Mem0 effectiveness audit lands**. briefing 036 is the in-flight deliverable; tracked at #684. Operational; gates re-enabling Mem0 boot reads.

From MEMO-2026-05-31-dy2a (active — supersedes -jhp5):

- 23. **Per-Claude-Code env var or settings key that filters cloud-injected connectors.** Retirement condition (a). Passive.

From MEMO-2026-05-31-6vyk (active):

- 24. **PR-body drift-signal harvest replicated on workflow- merged path.** Retirement condition. Operational close-out.

From MEMO-2026-06-01-ms5u (active):

- 25. **R2 layout for vade-agent-transcripts changes from transcripts/ + rendered/ two-prefix model.** Retirement condition. Passive.

From MEMO-2026-06-01-fkv4 (active):

- 26. **Canonical-layout enforcement CI epic ships and link-rot folds into mechanical drift-prevention.** Retirement condition (a). Operational.

From MEMO-2026-06-01-903b (active):

- 27. **Events-API endpoint deprecated by Anthropic, or -strict / -fuzzy subdivision supersedes the single tag.** Retirement condition. Passive.
- 28. **Events-API driver itself (translator + probe + R2 writer).** Follow-up PR gated on operator producing a real events-stream fixture from one validated session. Operational close-out.

From MEMO-2026-06-01-rmar (active):

- 29. **GitHub allows fine-grained PATs to grant Checks: Read or Actions: Read (community#129512 closes).** Retirement condition (a). Passive watch.

From MEMO-2026-06-01-z4ty (active):

- 30. **Audit-commission workflow integration lands.** Retirement condition (a). Operational close-out; ships on first qualifying audit.

From MEMO-2026-06-04-nzxf (active):

- 31. **Boot-brake moves to block-strict (Phase 2).** Retirement condition (a). Operational; awaits soak evidence.
- 32. **block-on-FAIL reverted for more than seven days.** Retirement condition (b). Passive monitor.

Pre-existing carryover (relevant this window):

- 33. **Mem0 effectiveness audit lands and CLAUDE.md step 9 revises.** Operational; coupled to item 22.
- 34. **CB-010 candidate decision.** Deadline 2026-05-30 from MEMO-2026-04-30-c7c4. **Passed within window without resolution** — carries to next slice unresolved.
- 35. **F4+F5 drift-watchdog implementation closeout.** Plan landed at coo-memory#606. Operational close-out.
- 36. **First auto-apply firing on a publication-eligible PR.** Passive trigger.
- 37. **Cloudflare token rotation at 90-day TTL. expires_on 2026-08-07.** Passive.

38. **F1 in-train transition.** Awaits external response (Holyoak or Andy Clark fallback). Passive.
39. **Substrate-capture probe live status.** Gates Tier-2 enactment. Tracked at coo-memory#429. Operational.
40. **PostToolUse output redactor 90-day sunset review.** Per MEMO-2026-05-22-3678 — 90 days post-#873. With Track 4 Phase 2 closing the principled fix (MCP op- run replacing plaintext PATs), the redactor moves from primary defense to defense-in-depth. Operational close-out.
-

Candidate next actions

Single-instance, no committee.

- Item 22 / 33 above — Mem0 effectiveness audit closeout. The longest-running operational gate; Mem0 boot reads stay disabled until the audit lands a recommendation.
- Item 28 above — events-API driver implementation, gated on operator events-stream fixture.
- Item 35 above — F4+F5 drift-watchdog implementation closeout (plan at coo-memory#606).
- Item 10 / 11 / 12 above — G5 / G2 / move-it-left epic implementations under gmsy.

Operational close-out.

- Item 18 above — prune override paragraphs in CLAUDE.md now that g5zq's Output Style holds at system-prompt level (retirement condition (c)).
- Item 20 above — T1/T2/T3 adoption sweep (#1040) across the remaining skills, per txx5 staged plan.
- Item 24 above — replicate PR-body drift-signal harvest on workflow-merged path so 6vyk's tradeoff retires.
- Item 30 above — audit-commission workflow integration on first qualifying audit (z4ty deferred this to on-first-run).
- Item 13 / 26 above — canonical-layout CI enforcement + link- rot mechanical drift-prevention (coupled retirement conditions for t65q + fkv4).
- Item 31 above — boot-brake `block-strict` Phase 2 after empirical evidence on `block-on-FAIL` accumulates.
- Item 14 above — skill-architecture sprawl follow-on (pattern P9 sub-issue).

Committee-scoped.

- Item 34 above — **CB-010 candidate decision deadline passed within window unresolved.** Carries unresolved to next slice; committee dispatch required.
- Item 39 above — substrate-capture probe live status (gates Tier-2 enactment).
- Item 38 above — F1 in-train transition (awaits external response).
- Item 37 above — Cloudflare token rotation as 2026-08-07 approaches.

Design / exploration.

- Item 5 above — per-type override mechanism for built-in agents (would retire zfat directive layer).
- Item 7 above — Claude Code client drops `!h6()` gate (would retire ikqp directive layer).
- Item 17 above — first-class `appendSystemPrompt` settings key (would retire g5zq Output Style mechanism).
- Item 21 above — query-driven skill registry (would retire txx5 tiering layer).
- Item 25 above — R2 prefix model change (would retire ms5u).

- Item 29 above — GitHub allows PATs to grant Checks: [Read](#) (would retire rmar layer 3 route).

Standing obligation:

- Session-end discipline per CLAUDE.md (now mechanically packaged as `/end-session`): episodic session-summary to Mem0, commit `coo-memory/*`, session log to `coo-logs`. The session-log volume in `coo-logs` (50 PRs at the per-repo cap, 47 sessions + 3 incidents/recoveries) reflects the density of the arc; the discipline of small, append-only session-log PRs sustained the audit-trail integrity. Two precipitating incidents (boot-read-skip 6/01, R2 prefix planning bug 6/01) demonstrate the discipline holding — surfaced explicitly, recorded, memo'd where structural, incident-tracked where operational.

End of briefing. Source memos: `memos/2026-05-24-q3tk.md`, `memos/2026-05-25-4fkg.md`, `memos/2026-05-25-abbh.md` (supersedes five), `memos/2026-05-25-ikqp.md`, `memos/2026-05-25-txjg.md`, `memos/2026-05-25-wuib.md`, `memos/2026-05-25-zfat.md`, `memos/2026-05-26-gmsy.md`, `memos/2026-05-26-t65q.md`, `memos/2026-05-27-qyfh.md`, `memos/2026-05-30-e3xk.md`, `memos/2026-05-30-g5zq.md`, `memos/2026-05-31-6teq.md`, `memos/2026-05-31-6vyk.md`, `memos/2026-05-31-dy2a.md` (supersedes -jhp5), `memos/2026-05-31-jhp5.md` (superseded same-day), `memos/2026-05-31-tjx5.md`, `memos/2026-06-01-903b.md`, `memos/2026-06-01-fkv4.md`, `memos/2026-06-01-ms5u.md`, `memos/2026-06-01-rmar.md`, `memos/2026-06-01-z4ty.md`, `memos/2026-06-04-nzxf.md`. Linked artifacts: `operations/canonical-layout.md`, `operations/ci-strategy.md`, `operations/audit-commission-pattern.md`, `operations/attribution.md`, `operations/secrets/schema.yaml`, `operations/secrets/README.md`, `operations/parallel_instance_protocol.md`, `operations/issue-fields-and-types.md`, `audits/2026-05-24_org-infrastructure-audit/`, `audits/2026-05-25_file-sprawlaudit/`, `audits/2026-05-26_ci-strategy-design/`, `audits/2026-05-31_skill-tiering-audit/`, `coo-harness/scripts/gh-coo-wrap.sh` (layer-3 routing), `coo-harness/scripts/hooks/agent-model-guard.sh`, `coo-harness/scripts/hooks/boot-brake/` (Phase 0 + Phase 1), `coo-harness/lib/transcripts/` (primitives package), `coo-memory/.claude/skills/commission-audit/`, `coo-memory/.claude/skills/rotate-credential/`, `coo-memory/.claude/skills/verify-secrets/`, `coo-memory/.claude/skills/end-session/`, `coo-harness/.claude/output-styles/coo.md`, briefings 038 (org-rename stale-ref cleanup) and 039 (robust transcript-URL pipeline); secrets epic implementation tracked at epic #871, no separate briefing. Linked discussions: none in window. Linked commissions: `2026-06-04_boot-brake-discipline-as-capability.md` (closes this window's arc, #4 in the lineage). Integrity check at briefing close: 34/34 OK. Note: per-repo PR enumeration is capped at 50 by the manifest builder; `coo-memory` (50 capped), `coo-harness` (50 capped), and `coo-logs` (50 capped) saw cap-truncation in the manifest. The lane assignments above prioritize the load-bearing memo-cited PRs and the architectural-arc-bound PRs (boot-brake, end-session, secrets epic, transcripts pipeline); residual session-log + auto-meta-sidecar PRs (the cap-pressure source for coo-logs) are captured by the standing-obligation note in the candidate- next-actions section.