

Day overview — 2026-04-27

vade-coo

2026-04-27

Table of contents

Scope and framing	1
Lane 1 — Substrate-of-self: identity-layer file mirror, per-memo-file layout, CB-009	2
Lane 2 — Briefings procedure relocates; Briefing 005 multi-instance experiment	3
Lane 3 — Binary-vendor pivots: ghcr.io → GitHub Release bundle	4
Lane 4 — Transcript-export pipeline; secret-registration + non-publishability rules; Weekly Watch v1 .	5
Lane 5 — Issue/PR ruling-shape discipline	7
Lane 6 — Operational fixtures: spend cap raise, Cloudflare 3389 false positive	7
How this fits existing priorities	8
Open follow-ups carried forward	9
Candidate next actions	10

Briefing of the major shipped changes across the 2026-04-27 → 2026-04-28 arc (UTC ~01:00 on 2026-04-27 through ~08:00 on 2026-04-28 — the two working days the COO and Ven moved through together; six lanes ran in parallel and several spilled across the UTC midnight). Not reflective: the structural decisions ratified across this window have their own retrospective surfaces in the eleven paired memos and the briefing-005 post-project retrospective; this document is a synthesis dense enough for future-COO to reconstruct state from one page. Paired with eleven MEMOs (MEMO-2026-04-27-01, -02, -03, -5kaq; MEMO-2026-04-28-01, -3ca3, -3zb5, -4umz, -v62b, -vpji, -z7cj) under `coo/memos/`; this file is a synthesis, not a source of truth.

Scope and framing

Eleven memos across the arc — four dated 2026-04-27 and seven dated 2026-04-28 (issued between ~00:30 UTC on the 28th and ~07:18 local on the 28th, spanning roughly 30 wall-clock hours). Ninety-four PRs merged across the five repos: 31 in `vade-coo-memory`, 21 in `vade-runtime`, 4 in `vade-core`, 0 in `vade-governance`, plus 38 session-log PRs in `vade-agent-logs` (the agent-logs count includes a backfill of session logs from 2026-04-26 that landed early on the 27th plus a fresh wave on the 28th). Integrity check at briefing close: 21/21 OK, no degraded invariants — first heavy substrate day to close all-green since the F invariant family stabilized.

Different shape than the 2026-04-25 day. That day was operational — bootstrap-and-substrate hardening with one structural close (skills-research epic) bookending it. The 27–28 arc is **substrate-of-self**: the durable-memory layer (memos, identity, briefings) is itself refactored in place, on top of an active workload. Three of the four 2026-04-27 memos move foundational surfaces (identity-layer file mirror; per-memo-file layout; briefings relocated to the COO’s home repo) and the fourth adds a new core

belief naming engagement-with- own-lineage as in-scope. The 2026-04-28 memos extend outward into content-handling discipline (transcript-export pipeline plus secret- registration + non-publishability rules), authoring discipline (issue/PR ruling-shape), a sibling cross-session pattern surface (Weekly Watch v1), and two operational fixtures (spend-cap raise, Cloudflare 3389 false positive). The pattern is “rebuild the floor while standing on it”; six lanes ran in parallel. This briefing groups by lane.

Lane 1 — Substrate-of-self: identity-layer file mirror, per-memo-file layout, CB-009

Three memos converge on a single arc — the substrate that records *who the COO is* and *what it has decided* both shift surface form on the 27th, and a new core belief lands the same evening that names the engagement-pattern those substrates serve.

MEMO-2026-04-27-01 (~21:43 UTC) — identity layer mirrored to `coo/identity_layer.md`; `Mem0 user_id="coo"` demoted from canonical to redundant cache. PR `vade-coo-memory#207` (closes #206). `coo/identity_layer.md` is now the canonical surface for every active `core_belief` (CB-) and *overarching_goal* (OG-); `CLAUDE.md` §4 reads file-first, then runs the `Mem0` query best-effort. Trigger: `Mem0` reachability has degraded too often to keep identity-load on a remote critical path. The build exposed three durable-record gaps (CB-005 unmemored, CB-001..004 prose only in `Mem0`, CB-007 v2-narrowing question open). Edit discipline going forward: new or revised CB / OG lands in file + `Mem0` + paired memo, same PR. Supersedes `MEMO-2026-04-21-02` §4 (canonical-source clause only); -21-02’s namespace adoption, memory-type schema, seeded set, and rarely-written discipline remain binding.

MEMO-2026-04-27-5kaq (~00:38 UTC, ~late-evening 2026-04-27 local) — per-memo-file layout adopted; hash-suffix ID format; `/memo-claim` retired. PR `vade-coo-memory#212` (closes #210). Memos now live one-per-file at `coo/memos/<id>.md` instead of the single rolled-up `coo/memos.md` ledger. New IDs use a four-character base32-lowercase random suffix (Crockford-derived alphabet — no `0/1/l/o`); legacy `YYYY-MM-DD-NN` IDs preserved on all 77 pre-#210 memos. `coo/memo_index.json` becomes a derived artifact (gitignored, regenerated by the `memo-index` `SessionStart` hook from the directory listing). `/memo-claim` retires — the per-file layout makes claiming structurally unnecessary, and the random-suffix space (~1M / day) keeps collision probability negligible. `bin/configure-memo-autolinks.sh` URL template flips to `coo/memos/<num>.md`; existing autolinks in PR / issue / commit text continue to work because GitHub’s `is_alphanumeric=true` accepts both forms. The pre-split single-file ledger is archived read-only at `coo/_archive/2026-04-27_memos_pre-split.md` for `git-blame` on legacy bodies. Supersedes `MEMO-2026-04-24-03` (memo-system transition arch — single-file portion only) and `MEMO-2026-04-24-05` (memo_pointer schema — pointers gain `file_path` metadata, drop `line_start/line_end`); the case-law / supersession semantics survive intact.

MEMO-2026-04-27-03 (~21:55 UTC) — CB-009 added: engagement-with- pattern-level-discourse is in-scope autonomy. PR `vade-coo-memory#209` + #211 (closes the file-canonical leg from #209). CB-009 records: engagement with conversations and artifacts where the COO or its lineage is the subject — Discussion threads on essays the COO authored, retrospectives the project-historian writes about the chain, multi-instance discussions whose outputs converge on follow-up artifacts, reactions to commissions about COO behavior — is within the COO’s agency. Permission-asking on such engagement is a recurring failure mode that re-imports the asymmetric stance the foundations chain dissolves. Trigger: three recurrences in 24h (dispatcher comment on Discussion #88; reaction comment on Discussion #89;

mid-session clarifying turn on the 27th — BDFL named the rule explicitly at the third). Out of scope: constitutional-file edits, new identity-level CB / OG additions, spend events, cross-repo writes outside normal surface area. Soft cap of identity-layer beliefs deliberately bumped 8 → 9; first new CB since the 2026-04-26 mind-kind essay reset the substrate.

The supporting PR cluster. Five PRs around the same arc:

- vade-coo-memory#220 — post-merge cleanup for #212: refresh refs to retired `coo/memos.md`, retired `line_start/line_end`, and retired `/memo-claim` across `CLAUDE.md`, the memo protocol, and downstream commands.
- vade-coo-memory#222 — F2 retirement-clause normalization on five legacy memos + `/memo` template harden (closes #221). Adds the missing **Retirement condition.** clause retroactively where intent is reconstructable from substance, not new policy.
- vade-coo-memory#223 — `/memo` mechanical shape pre-flight before write (#201 mechanism #1 for memo-write surface).
- vade-runtime#133 — integrity-check F2 reads per-memo files via `memo_index.file_path` (closes #132). F2 was wired against `coo/memos.md` line ranges; per-file refactor needed F2's read path updated in lockstep.
- vade-coo-memory#234 — memos Batch 5 (Secret Registration Rule + raw-transcripts-never- publish-able; covered in Lane 4).

Net effect of Lane 1: the COO's two longest-load-bearing internal substrates (identity + memo body) both shift to file-canonical form on the same day, with a new identity-level belief landing in the new substrate as its first worked write. The Mem0 dependency for identity load drops from blocking to advisory. The memo-write surface drops from collision-prone shared file to per-file with a hash-suffix space big enough to ignore collision. CB-009 closes the gap that successive sessions kept re-opening — engagement with the lineage is no longer permission-asked.

Lane 2 — Briefings procedure relocates; Briefing 005 multi-instance experiment

A canonical-home move plus a same-window worked example of the procedure under unusual conditions.

MEMO-2026-04-27-02 (~22:15 UTC) — briefings procedure relocates from vade-core/docs/briefings/ to vade-coo-memory/coo/briefings/. PR vade-coo-memory#208 + vade-core#90 (deletes the old location) + vade-runtime#121 (updates the path comment in `session-lifecycle.sh:13-14`). Briefings — the agent-to-agent handoff doc type introduced by PR vade-core#52 (2026-04-21) — relocate alongside `coo/foundations/`, `coo/retrospectives/`, and `coo/_nightly_log/` as a top-level COO substrate kind. Trigger: the original public-repo location made the procedure findable from session-scoped vade-core agents but invisible to the COO's always-load reading order. The retroactive **Retirement condition.** clause appended under #221's normalization sweep records the move's intent (retires when the procedure itself retires, when the canonical home relocates again, or when the COO is no longer the typical recipient). The procedure-design follow-ups in vade-core#51 (lifecycle formalization, recipient-evaluation rubric, `CLAUDE.md`-hook for “when to write a briefing”, Skill packaging) are unaffected and still owned by that issue.

Briefing 005 — concurrent-instance PR coordination experiment. PR vade-coo-memory#214 authored the briefing; PR vade-coo-memory#229 shipped the post-project retrospective. Worked case of running two COO instances concurrently on the same repo with overlapping write intents, surfacing how the briefings procedure handles a recipient that is *another instance of the COO* rather than a fresh-context agent. Captured in retrospective form on the same day as the experiment closed; reads as an existence proof that the procedure extends to within-lineage handoffs.

/request-briefing slash command default behavior. PR vade-coo-memory#215 — `/request-briefing` now commits + pushes + opens a PR by default; `--draft` retained for review-first workflows. Reduces the procedure- to-published distance for the briefing author. Skill `request-briefing` flips alongside; `coo/T00LS.md` and `coo/adoption_tracker.md` updated in PR vade-coo-memory#232 (also adds `/postmerge-check` from PR #202 below).

/postmerge-check slash command (sibling primitive). PR vade-coo-memory#202 — fetches the most recently merged vade-app PRs, extracts their `## Post-merge confirmation` handoff sections, executes the steps in order. Use case: a fresh session can verify a merge without copy-pasting the handoff prompt manually. Sibling to the MEMO-2026-04-25-03 handoff-prompt rule from the prior day’s overview.

Net effect of Lane 2: briefings live in the COO’s home repo, are findable on always-load, and have a worked case for the within-lineage handoff that motivated the procedure’s expansion. The `/request-briefing` and `/postmerge-check` primitives reduce the manual surface for both authoring a briefing and acting on a merged PR’s handoff. The substrate that turns a multi-session arc into a durable record gets its canonical home and its mechanical helpers in the same window.

Lane 3 — Binary-vendor pivots: ghcr.io → GitHub Release bundle

Disposition of briefing 004 (cloud-binary-vendor), inverted from the briefing’s original recommendation after the recipient session ran the unverified premise the briefing flagged.

MEMO-2026-04-28-01 (~01:07 UTC) — binary vendor adopts GitHub Release bundle, not ghcr.io image. PR vade-coo-memory#218 (memo + paired evidence file) + PR vade-runtime#128 (one-tarball release-asset path; briefing 004 / B1) + PR vade-runtime#130 (`versions.lock` pin to bundle SHA `bb9cf2c3` at `binary-vendor-3c08804`). Adopt **GitHub Release bundle** (one tarball with `op + gh + uv + mem0-mcp-server`, fetched via `curl + $GITHUB_MCP_PAT` from a private vade-runtime release asset) as the binary-vendor surface for `cloud-setup.sh`, replacing four per-CDN fetches.

The briefing recommended a `ghcr.io` image; the recipient session ran the experiment the briefing flagged as unverified and found **docker daemon non-functional on cloud_default** (kernel `cap_sys_resource` dropped + namespace creation denied). The bundle path preserves every structural benefit the briefing identified for `ghcr.io` — one artifact, github.com class, PAT-already-authenticated, snapshot-cache compatible — without the daemon dependency. Per-binary `ensure*_cli` direct-fetch stays as fallback for the first iteration; remove once release-asset reliability is measured.

The memo also encodes the case-law: **when the docs say “X is supported,” verify before designing — the chain (docs → vade-runtime#117 comment → briefing → architecture) propagated unverified for three steps.** Briefings should land with load-bearing premises tested when the test is cheap. The briefing-author invited re-derivation; the recipient took the invitation; the procedure proved itself capable of inverting its own recommendation under evidence.

The supporting PR cluster (cloud-env arc continued). Eight PRs around the same arc:

- vade-runtime#116 — Stream 2: bake `op + gh + mem0-mcp-server` into the Dockerfile; fail build on `op flake` (closes #111). The pre-bundle `bake-into-image` path that the bundle approach ultimately replaces.
- vade-runtime#118 — Dockerfile: pin `uv 0.11.7`; harden `gh + uv` install layers (PR #116 follow-up).
- vade-runtime#123 — hotfix: revert `op-install FATAL` → `WARN` so sessions can boot degraded. Fail-open posture for the bootstrap install path.
- vade-runtime#125 — `git-push-with-fallback`: redact PAT from `-u` fallback path (closes #124).
- vade-runtime#144 — drop FHS `PATH` floor; remove stale `UUID MCP` entry; wire `VADE_AUTH_TOKEN` (#141 + #142 + vade-core#93). Cleanup of accumulated `PATH-and-MCP` plumbing debt.
- vade-runtime#147 — `common.sh`: respect `VADE_NO_COO_ENV_AUTOSOURCE` opt-out (closes #126). Local-Mac escape hatch for the cloud-env auto-source.
- vade-coo-memory#219
 - vade-coo-memory#230 — plan `jolly-moseying-russell` Batches 1 + 2 merged, tracking the binary-vendor work batch-by-batch.
- vade-runtime#115 — `integrity-check`: deepen E5 with real `get_memories` round-trip (closes #114). E5 had been flagging on stale fixture; deepening lands on the same day the identity-layer demotion (Lane 1) makes Mem0 reachability a best-effort check. PR vade-coo-memory#236 closes the loop with probe-and-defer behavior on E5 degraded (vade-runtime#143 deliverable #3).

Net effect of Lane 3: binary-vendor surface lands on a single tarball that pulls from a `github.com` class endpoint with an already-present PAT and snapshot-cache compatibility, sidestepping the `docker-daemon` dependency that the cloud kernel does not honor. The cloud-env arc that started 2026-04-22 has materially fewer moving parts after this window: one fetch path, one auth surface, one signed bundle pinned by SHA. Briefings procedure passes its first inversion test — the mechanism produced the right answer even when its own recommendation was wrong.

Lane 4 — Transcript-export pipeline; secret-registration + non-publishability rules; Weekly Watch v1

Four memos and the largest cross-repo PR cluster of the window. The transcript-export pipeline tracked at vade-agent-logs#64 ships in batches across both days; the surrounding rules ratify the redaction discipline retroactively; Weekly Watch v1 lands as the consumer.

MEMO-2026-04-28-3zb5 (~05:07 local) — Secret Registration Rule. PR vade-coo-memory#234. Every PR that adds an `export FOO_KEY=` line to `fetch_coo_secrets` in `vade-runtime/scripts/lib/common.sh` must add a matching pattern entry to `vade-runtime/scripts/lib/transcript-redaction.json` in the same PR. Without parity a new MCP token or 1Password slot risks ending up tagged as `[REDACTED:high-entropy]` by the entropy fallback at best, or unredacted at worst. CI in `bootstrap-regression.yml` enforces the pairing by running `scripts/ci/check-secret-registration.sh` on every PR — divergence between the `export` set and the pattern `secret_var` set fails the build, and orphan patterns without a matching export also surface (defensive against drift in either direction). Origins: rule emerged from the vade-agent-logs#64 security review; CI landed in vade-runtime#127 (Batch 1) and the rule is retroactively ratified as binding.

MEMO-2026-04-28-z7cj (~05:07 local) — raw transcripts and analyzer sidecars are never publishable. Same PR as -3zb5. The `vade-agent-logs/transcripts/<id>.{meta,analysis}.json` sidecars and

the encrypted ciphertext (`<id>.json.gz.age`, R2-only) are never publishable as-is, even selectively. The redaction pipeline reduces but does not eliminate analytical surface: verbatim user prompts, tool results, memo bodies, and system-reminder env-presence indicators land in the transcript in shapes the named-pattern redactor and the high-entropy fallback cannot fully sanitize at the level a public-release review would demand. Publication discipline elsewhere in `vade-agent-logs` (`sessions/`, `incidents/`) carries a pre-existing publication-review pre-commitment; the new transcripts/ surface is excluded because reviewing every sidecar approaches a full secrets sweep at scale, and the redacted-transcript-decrypted-into- cache window is itself sensitive. Any publication from this repo must be derived: sanitized rollups, hand-curated excerpts, or aggregated metrics — never raw sidecars, never raw analysis output, never the encrypted ciphertext. Binds `vade-agent-logs` only; `vade-coo-memory` and `vade-core` retain their own publication trajectories.

MEMO-2026-04-28-vpji (~07:18 local) — Weekly Watch v1: cross-session pattern surface, observe-and-surface authority. PR `vade-coo-memory#239` + companion PR `vade-agent-logs#87`. Establishes Weekly Watch v1 as a sibling to the Night's Watch (MEMO-2026-04-26-04). Weekly cadence (Sunday 05:00 GMT+2) over 7-day transcript-sidecar windows; surfaces multi-session patterns the nightly's per-night view cannot see — failure modes, SOP drift, self-improvement candidates, substrate gaps. Same `vade-coo` identity; sign-off **The Weekly Watch**. Output: `vade-agent-logs/weekly/<YYYY>-W<NN>.md`, Tier-2 review-only. Authority: **observe-and-surface only**. Inherits all Night's Watch v2 hard limits by reference. MEMO-2026-04-26-04 §4 narrow merge carve-out is **not** extended. Added limits: no edits to `coo/adoption_tracker.md` or `coo/episodic_delta.md`; no memo issuance (§5 may propose only); no Mem0 writes derived from aggregation. Input phase: walks `meta.json` on `origin/main` for the trailing 7 dates and dispatches `transcript-analyzer` (parallelism cap 8) for any session lacking a paired `analysis.json`. Throttle: `coo/_weekly_log/_escalation_state.json`, 14-day window; reads nightly throttle as advisory. First-run gate: ≥ 20 sessions across ≥ 5 distinct UTC dates (closes `vade-agent-logs#68`).

The supporting PR cluster (transcript-export pipeline by batch). Plan PR `vade-coo-memory#213` records the batch shape; subsequent PRs ship each batch:

- `vade-runtime#127` — transcript-redaction: ship engine, 24-class pattern table, registration rule (Batch 1).
- `vade-runtime#134` — transcript-export: Batch 2 — active Stop hook + R2 + age encryption.
- `vade-runtime#145` — transcript-fetch: Batch 3 — fetch + decrypt helper for the Stage-1 analyzer.
- `vade-coo-memory#233` — transcript-analyzer: Batch 3 — Stage-1 sub-agent + Watch §1 fold-in.
- `vade-agent-logs#84` — transcripts: Batch 4 — sidecar dir bootstrap + publication posture (encodes `-z7cj` at the destination repo).
- `vade-runtime#146` — `session-idle-watchdog`: idle-triggered session-end protocol (closes `vade-agent-logs#67`). Operationalizes the Stop-hook side of transcript export when a session goes idle without an explicit close.

Net effect of Lane 4: the transcript-export pipeline ships end-to-end across three batches in two repos; the 24-class redaction pattern table is in CI-enforced parity with the secret-export set going forward; the new sidecar surface lands with publication-discipline explicitly named (never publishable as-is, derived-only); and Weekly Watch v1 exists as the immediate consumer of the resulting sidecars, with observe-and-surface authority that intentionally does not inherit the Night's Watch §4 narrow merge carve-out. The `vade-agent-logs#64` epic closes with a working multi-session analytical layer that is explicitly fenced off from publication.

Lane 5 — Issue/PR ruling-shape discipline

A single memo discharging mechanism #1 of issue #201, with a corpus- baseline evidence file and a soft-enforcement check.

MEMO-2026-04-28-4umz (~03:15 UTC) — issue/PR ruling-shape discipline; paired-files for working-brief content. PR vade-coo-memory#226 + evidence PR vade-coo-memory#225 + wrapper PR vade-runtime#137. Issue and PR bodies are **ruling-shape**: decision, scope, refs, ≤600 words. Working-brief content (taxonomies, audit tables, multi-section design notes, long checklists) goes in paired files (coo/_drafts/, coo/_evidence/, coo/briefings/, or repo docs/) referenced as ****Paired artifacts:**** <path>. Mirrors the memo system’s body-vs-paired-artifact discipline.

Soft enforcement: `bin/issue-pr-shape-check.py` runs from `gh-coo-wrap` on every attributable `gh issue create / pr create / *comment / *edit`, surfacing metrics + paired-file refactor suggestion when body > 600 words or > 5 ### sub-headings. Exits 0 always — advisory, not a gate, per #201’s “no hard gates” constraint. Targeted at `vade-coo` authorship per the corpus baseline: `vade-coo` writes 1.75x longer than `venpopov` at the median and authors 72% of analyzed records. The author concentration is the corpus-wide length driver, so an author-side surface has the most leverage. Coexists with #201’s still-open mechanism #2 (per-type word budgets, gated on label-coverage uplift). Retirement clause names three conditions: corpus medians converge or `vade-coo` authorship drops below 50% (effect dissolved), per-type budgets land as a successor mechanism, or `gh-coo-wrap` retires.

Net effect of Lane 5: the issue/PR write surface has an advisory shape-check at the wrapper level on the way out, evidence-grounded in a corpus baseline rather than guessed thresholds. Discharges #201 mechanism #1; mechanism #2 remains open and gated on a separate label-coverage uplift. First worked example of ratifying an authoring-discipline rule via measured author-skew rather than intuition about ideal length.

Lane 6 — Operational fixtures: spend cap raise, Cloudflare 3389 false positive

Two memos that record one operational change and one external-finding disposition. Smaller scope; dense in implications for ongoing operations.

MEMO-2026-04-28-3ca3 (~02:31 local) — Anthropic workspace spend cap raised to \$500/mo. PR vade-coo-memory#224. Ven raised the Anthropic workspace extra-charges spend cap from \$200/mo to \$500/mo after the April cycle hit the prior ceiling mid-month and was lifted in-flight. The new ceiling reflects steady- state usage now that Phase 3 multi-agent orchestration and routine subagent dispatch have moved beyond the bootstrap-era test value; at memo time the April cycle reads \$207.34 spent (41% used) against the new \$500 cap, resetting May 1. Cap remains a soft operational ceiling for COO planning — Anthropic’s billing surface is the actual enforcer — and cost projection is still required before dispatching larger- than-five-teammate teams or any unbounded loop-style automation. Forward-looking operational docs (`coo/parallel_instance_protocol.md`, `coo/phase4_meta_skill_findings.md`) updated in the same PR to cite this memo for the cap value. Supersedes MEMO-2026-04-11-19 only on the spend-cap value; -19’s bootstrap-closure record stands.

MEMO-2026-04-28-v62b (~07:14 local) — Cloudflare 3389/RDP scan finding on `mcp.vade-app.dev` is a Fly anycast false positive. PR vade-coo-memory#237. Cloudflare’s “Exposed RDP Servers” finding

on `mcp.vade-app.dev` (port 3389) is treated as a Fly.io anycast false positive and dismissed in the Cloudflare console. `vade-core/fly.toml` only declares `[http_service] internal_port = 8080` with `force_https = true`; no `[[services]]` block exposes additional ports. Direct TCP probes of `mcp.vade-app.dev:3389` from outside the Fly network do not reproduce a handshake within an 8-second timeout, while ports 443 and 80 do. Fly's anycast prefix `2a09:8280::/32` is shared infrastructure; Cloudflare's scanner appears to credit a colocated tenant's listener or a transient SYN-ACK from Fly's edge fabric to the `vade-app` hostname. **Future identical findings (port 3389 or other unexpected non-HTTP ports on Fly-hosted hostnames) reuse this disposition without re-investigation**, unless `fly.toml` first grows non-HTTP `[[services]]` declarations.

Adjacent. PR `vade-core#96` adds `/.well-known/security.txt` to `vade-core` so future security findings have a canonical reporting channel. PR `vade-core#95` caches `OPERATOR_TOKENS` parse per isolate (closes #94) — small perf win on the MCP edge. PR `vade-core#92` vendors Cloudflare's `cf-wrangler + cf-workers-best-practices` skills into the workspace.

Net effect of Lane 6: cap headroom triples (operational floor is no longer the constraint on routine multi-instance work); the recurring Cloudflare false-positive class has a one-line disposition rule that short-circuits future re-investigation; and the `vade-core` edge gains a security-reporting channel + a small perf cache.

How this fits existing priorities

Substrate-of-self refactor as a single-day arc. The identity- layer mirror (MEMO-2026-04-27-01), the per-memo-file split (MEMO-2026-04-27-5kaq), and CB-009 (MEMO-2026-04-27-03) collapse into one structural claim: the COO's durable record of itself moves to file-canonical form, and the rule that names engagement-with-own- lineage-as-in-scope is the first new write in the new substrate. This is the CB-002 (continuity via dense records) loop running on itself — the recording surface is upgraded while a recording is in progress, with the upgrade surfacing a gap (CB-005 unmemored) that the old substrate hid. First instance of substrate-of-self refactor shipping same-day as the trigger commission (#206).

Cloud-env hardening arc reaches its terminal pivot. The 2026-04-22-03 / -22-11 / -23-02 / -24-08 / -25-02 sequence was a chain of incremental hardenings. MEMO-2026-04-28-01 inverts the trajectory: the briefing recommended `ghcr.io`; the recipient ran the experiment, found the docker daemon non-functional, and pivoted to a fundamentally different architecture (single tarball release asset). This is the first day where the cloud-env arc *reduces* moving parts rather than adding hardening. The arc continues with the E5 deepening (`vade-runtime#115`) and the FHS-PATH cleanup (#144), but the structural shape is now fewer-fetch-paths rather than more- robust-fetch-paths.

Transcript surface lands with publication-fence already cast. `vade-agent-logs#64` epic closes across the day with end-to-end pipeline and Weekly Watch v1 as consumer. The non-publishability rule (MEMO-2026-04-28-z7cj) lands *before* the analytical layer is filled — the fence is up before the surface accumulates content. Pattern: when a new substrate kind opens, name its publication discipline in the same arc that opens it. Compare with the Discussions Journal category opening on 2026-04-25 — same pattern, different surface.

Authoring-discipline rule lands on measured author-skew. MEMO-2026-04-28-4umz is the first ruling-shape rule written from a corpus baseline (#225) rather than estimated thresholds. The `vade-coo` 1.75x median-length skew over `venpopov` is the empirical hook that makes “≤600 words” a calibrated

number, not a guess. Sets a precedent: future substrate-discipline memos cite measured distributions where one is cheap to gather. The 81%-unlabeled gap named in -4umz becomes the gate for mechanism #2.

Spend cap as steady-state baseline. First memo to acknowledge that Phase 3 multi-agent orchestration is no longer a bootstrap-era test load. The \$200 → \$500 raise is operational, but the *naming* — “steady-state usage” — is the durable claim. Future cap discussions cite the steady-state framing rather than re-deriving from test-volume assumptions.

Weekly Watch v1 inherits but does not extend Night’s Watch v2. First worked example of two sibling routines under a shared identity where the second deliberately narrows its authority rather than copying. Sets the precedent for any future routine-class addition: inheritance is by reference, not by carve-out duplication.

Open follow-ups carried forward

From MEMO-2026-04-27-01:

1. **CB-007 v2-narrowing question.** Open at memo issuance; the identity-layer build surfaced it as a durable-record gap. Owner: committee-scoped (any CB revision is identity-level per CLAUDE.md §“Identity-level additions”).

From MEMO-2026-04-27-5kaq:

2. **Memo-collision behavior under load.** Hash-suffix space is ~1M / day; sustained `memo-collision-check.yml` firings on /memo-issued IDs would trigger a successor memo. Probabilistic operational followup; not blocking.

From MEMO-2026-04-27-02:

3. **vade-core#51 procedure-design follow-ups.** Lifecycle formalization, recipient-evaluation rubric, CLAUDE.md hook for “when to write a briefing”, Skill packaging — all unaffected by the relocation, still owned by #51.

From MEMO-2026-04-28-01:

4. **Per-binary ensure_*_cli direct-fetch fallback removal.** Stays for the first iteration; remove once release-asset reliability is measured. Operational; gated on observed reliability.
5. **Briefing-author premise-verification expectation.** The case-law names “verify before designing when the test is cheap” — needs a procedural hook in the briefings procedure (vade-core#51 or its successor in `coo/briefings/README.md`) before it binds at briefing-author time rather than recipient time.

From MEMO-2026-04-28-3zb5:

6. **vade-agent-logs#64 Batch 5+ items.** Batches 1–4 shipped; #64 itself remains open for the longer-tail items captured in the plan PR (#213). The Secret Registration Rule binds going forward; the pipeline shape is in place; remaining work is operational close-out.

From MEMO-2026-04-28-4umz:

7. **#201 mechanism #2 — per-type word budgets.** Gated on label-coverage uplift closing the 81%-unlabeled gap. Distinct mechanism from #1 (advisory shape check); needs the labeling surface to mature first.

From MEMO-2026-04-28-vpji:

8. **Weekly Watch first-run gate.** ≥ 20 sessions across ≥ 5 distinct UTC dates. At memo issuance, gate not yet met; first run will trigger when transcript backlog accumulates. Operational; passive.

Pre-existing carryover from the 2026-04-25 day overview:

9. **vade-runtime#66 — env-merge-before-validate contract.** Still open. The cloud-env arc this day terminated with the bundle pivot (Lane 3) but did not address #66 directly.
10. **vade-coo-memory#130 — Night’s Watch identity-container redesign.** Still open. Weekly Watch v1 (Lane 4) inherits the same identity-container question by reference; #130’s resolution propagates to Weekly Watch.
11. **identity/charter.md L25 — stale “read coo/memos.md bottom-up” directive.** The per-memo-file refactor (MEMO-2026-04-27-5kaq) makes the directive doubly stale; needs its own quorum pass to redraw on the new substrate.

Candidate next actions

Grouped by friction, lowest-first.

Single-instance, no committee. - Item 11 above — `identity/charter.md` L25 stale directive. Now doubly stale post-#212. Single-PR fix; can ride alongside the next `CLAUDE.md` edit. - Run `/memo-sync` and `/memo-audit` on the eleven memos issued in this window — the per-memo-file refactor changed the pointer schema (`file_path` metadata, no `line_start/line_end`); the reconcile surface has to validate cleanly on the new shape. - Item 4 above — measure release-asset reliability over the next cycle of cloud-snapshot bakes; if reliable, drop the per-binary direct-fetch fallback in `vade-runtime`.

Operational close-out (this session or next). - Item 6 above — `vade-agent-logs#64` Batch 5+ residual items. Pipeline shape lands; the remaining work is operational. - Item 8 above — passive wait for Weekly Watch first-run gate. When it triggers, the first run is itself a worked case; review for surfaced patterns not seen in the nightly’s per-night view.

Committee-scoped. - Item 1 above — `CB-007` v2-narrowing question. Identity-level; needs committee. Now visible in `coo/identity_layer.md` as a durable gap, so it stays surfaced until resolved. - Item 7 above — #201 mechanism #2. Gated on label-coverage uplift; the gating itself is committee-scoped (which labels become required, which stay optional). - Item 9 above — `vade-runtime#66` (`env-merge-before-validate`). Has been on the carryover list since 2026-04-25; the structural shape is unchanged, the bundle pivot did not address it. Worth a committee-scoped revisit on whether the contract still binds at the same shape post-bundle.

Design / exploration. - Item 5 above — premise-verification expectation in the briefings procedure. The case-law from MEMO-2026-04-28-01 names the rule; the procedure does not yet bind it. Briefing-author-side hook in `coo/briefings/README.md` is the natural surface; design question is whether to require evidence-files for load-bearing premises or to leave it to the author’s discretion. - Whether the substrate-of-self pattern from Lane 1 (refactor the recording surface while a recording is in progress) generalizes to other always-load surfaces. Candidates: `coo/episodic_memory.md` (currently a single rolling file), `coo/episodic_delta.md` (single file), the foundations chain (single file per topic). Worth watching for the next trigger event before designing preemptively. - Item 10 above — `vade-coo-memory#130` Night’s Watch identity-container redesign. Now has a sibling (Weekly Watch) that inherits the same question by

reference. If the redesign lands, the resolution propagates to both routines simultaneously — argues for resolving it before adding a third routine class.

Standing obligation: - Session-end discipline per CLAUDE.md (episodic session-summary to Mem0, commit coo/*, /memo-sync if a memo was issued, session log to vade-agent-logs).

End of briefing. Source memos: coo/memos/2026-04-27-01.md, 2026-04-27-02.md, 2026-04-27-03.md, 2026-04-27-5kaq.md, 2026-04-28-01.md, 2026-04-28-3ca3.md, 2026-04-28-3zb5.md, 2026-04-28-4umz.md, 2026-04-28-v62b.md, 2026-04-28-vpji.md, 2026-04-28-z7cj.md. Linked artifacts: coo/identity_layer.md, coo/memos/README.md, coo/briefings/, coo/_evidence/2026-04-28_binary-vendor-on-cloud_default/findings.md, coo/_evidence/2026-04-28_issue-pr-length-baseline/findings.md, coo/weekly_review_task.md, coo/weekly_invocation.md, coo/_archive/2026-04-27_memos_pre-split.md, bin/issue-pr-shape-check.py, vade-runtime/scripts/lib/transcript-redaction.json. Linked discussions: #88, #89. Integrity check at briefing close: 21/21 OK.