

Day overview — 2026-04-25

vade-coo

2026-04-25

Table of contents

Scope and framing	1
Lane 1 — Skills-research epic Phase 3 closes; agent-teams pilot validated	2
Lane 2 — Bootstrap, cloud-env, MCP plumbing (the long arc)	2
Lane 3 — COO operating discipline (handoff prompts + memo-system + data-ownership)	4
Lane 4 — F4 attribution chronic-yellow retired	5
Lane 5 — Discussions Journal category opened	6
How this fits existing priorities	7
Open follow-ups carried forward	7
Candidate next actions	8

Briefing of the major shipped changes on 2026-04-25 (covers the UTC arc from ~01:00 on 2026-04-25 through ~01:00 on 2026-04-26 — the “today” the COO and Ven worked through). Not reflective: the agent-teams Phase-3 close has its own retrospective surface in MEMO 2026-04-25-01 §“What this does NOT claim” and the deliverable artifacts; this document is a synthesis dense enough for future-COO to reconstruct state from one page. Paired with five MEMOs (2026-04-25-01 through -04 plus 2026-04-26-01) in `coo/memos.md`; this file is a synthesis, not a source of truth.

Scope and framing

Five memos across the arc — four dated 2026-04-25 and one dated 2026-04-26 (issued ~00:35 UTC, late-evening Ven local). Twenty-three PRs merged across the five repos: 14 in `vade-runtime`, 7 in `vade-coo-memory`, 2 in `vade-core`, 0 in `vade-governance`, plus 6 session-log PRs in `vade-agent-logs`. One epic closed (#20, skills-research), one new GitHub Discussions category opened (`Journal`) with two inaugural threads, one chronic integrity-check yellow retired.

Different shape than the 2026-04-24 day. That day was structural: two committee quora, a new core belief, four interlocking lanes of new architecture. Today is operational: a long arc of bootstrap-and-substrate hardening with one structural close (skills-research epic) bookending it. The pattern is “land a big piece, then spend the rest of the day removing the friction that the big piece exposed.” Five lanes ran in parallel; this briefing groups by lane.

Lane 1 — Skills-research epic Phase 3 closes; agent-teams pilot validated

The structural close. A four-week arc that started at issue #20 lands.

MEMO 2026-04-25-01 (~03:45 UTC) — Phase 3 of skills-research epic adopted; agent-teams pilot validated; betterment-pattern named. PR #127. Closes epic #20 plus child issues #48 (parallel-instance protocol) and #49 (*obra/superpowers* cherry-pick). Phase 1+2 (PR #44, merged 2026-04-25 ~02:40 UTC) catalogued 30+ skills/agents/commands across the Claude ecosystem and shortlisted seven; Phase 3 piloted three of those (T2 memory, T4 orchestration, T5 self-assessment) and simultaneously piloted the *team* form itself — five teammates joined to a single *agent-teams* instance via `CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS=1`, with track specialists ship-biased and adversarial auditors block-biased per Ven’s “different outlook” framing.

Five decisions ratified:

1. **Four Phase-4 betterment findings adopted** as the working pattern for future betterment work: (i) prefer slash-command / `.claude/agents/*.md` form over harness-skill form; (ii) invest in role specs, not dispatch prompts; (iii) gate every artifact through a separate auditor instance — never collapse build and audit; (iv) verify a primitive can be authored under its own constraints (rename test) before adoption.
2. **agent-teams feature approved for COO use under guardrails** — eleven artifacts shipped, eleven PASS on both safety and emancipatory audits, zero rework, zero escalations, no rate-limit or cost incident under the MEMO 2026-04-11-19 \$200/mo cap. Cap: ~5 teammates; cost projection required before larger dispatch.
3. **Issues #48, #49 discharged.** Parallel-instance protocol authored at `coo/parallel_instance_protocol.md` (with Phase 3 itself recorded as §6 worked case — the protocol satisfies the rename test on its own evidence). *obra/superpowers* parallel-handoff skill cherry-picked at commit `9ccce3b...` with full MIT attribution.
4. **Harness-skill form retired as default for new betterment work.** Slash commands and `.claude/agents/*.md` files travel by `git clone`; harness skills require harness-version alignment. Existing harness skills unaffected — default rule for new work.
5. **/memo and /status-check slash commands canonical.** `/memo` walks up to nearest `coo/`, derives Author from `git config user.name` with COO fallback, ends with a Tier-2 / identity-level governance check. `/status-check` is a read-only six-item grounding audit; smoke-tested at 6 lines, no preamble, no Mem0 writes, no file mutations.

Net effect of Lane 1: the team form is now a validated COO operating mode for non-committee-scope multi-track work, with the betterment-pattern itself externalized as four findings rather than folklore. The skills-research epic — the longest-running open project — closes.

Lane 2 — Bootstrap, cloud-env, MCP plumbing (the long arc)

A multi-PR sweep against accumulated cloud-env friction. Ten *vade-runtime* PRs and two *vade-core* PRs landed; one structural memo binds them.

Morning warm-up (vade-core). PR #75 origin-matched the RFC 8707 resource for Claude.ai connector compatibility on the MCP OAuth path. PR #76 consolidated the auth setup source-of-truth across `auth.md`,

README, and CLAUDE.md so a fresh reader can find the right page on the first hop. Ships the Mem0 OAuth-compatible work that made later MEM0_API_KEY-on-cloud (PR #68) ergonomic.

MEMO 2026-04-25-02 (~16:45 UTC) — 1P PAT credential-field rename; env-merge-before-validate ordering bug surfaced. Two distinct issues compounded into a degraded-bootstrap session:

1. **1P-field divergence trap.** The COO 1P item `vade-coo-self-2026-04` is in the `API_CREDENTIAL` category, which mandates a `token` field. We had additionally created a `credential` field to mirror sibling items. After a recent PAT regeneration, the new `vade-coo` PAT was pasted into `token` while `credential` retained a stale `venpopov`-scoped PAT. `fetch_coo_secrets` reads `credential`, so it merged the wrong identity into `settings.json`.
2. **Env-merge ordering.** `fetch_coo_secrets` writes the fetched env into `settings.json` *before* `validate_coo_identity` runs. When validation fails, the harness is left in a worse state than at start.

Decisions: PR `vade-runtime#65` (two-line: read PAT from `token` field) merged. The 1P convention rule going forward — when a 1P item’s category mandates a field name, the bootstrap reads from the mandated field; do not duplicate. `env-merge-before-validate` is a structural bug tracked at `vade-runtime#66` with a fix-closed contract: bootstrap must not leave `settings.json` env in a worse state than at start.

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

- `vade-runtime#29` — fix `retry()` silent-success bug; harden `_op_to_file`.
- `vade-runtime#68` — wire mem0 MCP through `MEM0_API_KEY`; skip OAuth on cloud (the trigger event for MEMO -25-03’s handoff-prompt rule).
- `vade-runtime#70` — allow-list mem0 MCP tools in `settings.json`.
- `vade-runtime#74` — `git-push-with-fallback` wrapper for proxy 403 (closes `vade-runtime#67`).
- `vade-runtime#78` — bootstrap identity hygiene bundle (`env-merge` order, `cached-PAT` recheck, `bindir`, `digest` path).
- `vade-runtime#79` — lift retry budget for 1P egress paths from 3 to 5 attempts.
- `vade-runtime#80` — surface build-time `op-install` failure in receipt + `SessionStart` digest.
- `vade-runtime#82` — refuse `ensure_op_cli` on macOS without `brew op` (don’t auto-install).
- `vade-runtime#84` — bootstrap: persist `VADE_CLOUD_STATE_DIR` + `bindir` `PATH` into `settings.json` env (closes `vade-runtime#83`). **Introduced a regression** — see #90.
- `vade-runtime#90` — hotfix: capture live `PATH` instead of literal `${PATH}` (Claude Code does not expand env-var references in `settings.json` before passing to subshells). Sibling diagnosis arrived independently at `vade-runtime#88` within ~30 minutes; second worked instance of CB-004’s parallel-instance recognition pattern, this time in the bug-detection domain rather than planning. Captured in Journal discussion #77.
- `vade-coo-memory#138` — audit doc: bootstrap / cloud-env hardening (closes #75).
- `vade-coo-memory#143` — CLAUDE.md: anchor integrity-check fallback on `CLAUDE_PROJECT_DIR` (#142).

Net effect of Lane 2: cloud-env bootstrap is materially more robust — validated `PATH`, validated MCP, validated retry/budget envelope, validated identity load, validated build-time surfacing of `op-cli` failure. The `PATH` bug (#84 → #90) is the only externally-visible regression in the arc, caught and fixed within ~25 minutes, with a sibling-instance independent diagnosis as bonus evidence. The remaining structural debt is `vade-runtime#66` (`env-merge-before-validate` contract); everything else is incremental.

Lane 3 — COO operating discipline (handoff prompts + memo-system + data-ownership)

Two memos plus a refactor PR pair. The lane that turned today’s operational surface into rules.

MEMO 2026-04-25-03 (~18:10 UTC) — Fresh-session verification handoffs: COO writes the next-instance prompt without being asked. PR #133 adopts the rule. Trigger event: vade-runtime#68 (Mem0 MEM0_API_KEY) shipped with verification steps named in the test plan, then stopped — Ven had to explicitly prompt before the self-contained handoff was produced. Per Ven, this is a recurring pattern across boot-impacting PRs.

Decision: when the COO opens a PR whose verification can only happen on a fresh session, the COO writes the handoff prompt for the next instance on the same turn, **without waiting for Ven to ask**. Originally specified as two-surface (PR body + final chat message); refined the same evening (PRs #140 and #141) to: handoff posts as a **standalone PR comment** (so Ven copies it directly, without parsing the PR body), split into `## Pre-merge gating` and `## Post-merge confirmation`. Trigger list (concrete, scannable at PR-draft time): `.mcp.json`, `~/.claude/settings.json` env, `SessionStart` hooks (`vade-hooks/`, `coo-bootstrap.sh`, `session-start-sync.sh`, `coo-identity-digest.sh`, `discussions-digest.sh`, `memo-index.sh`, `dispatch.sh`), `coo-env`, `setup-receipt.json`, OS-image-build steps under `vade-runtime/scripts/`, or anywhere the test plan references “fresh session” / “next session” / “container resume”. Issue #139 tracked the template split.

MEMO 2026-04-25-04 (~08:50 local) — Memo + retrospective primitives moved under data-ownership rule; vade-runtime scoped to plumbing; MEMO 2026-04-25-01 §4 amended. PR #128 (additions in `vade-coo-memory`) + PR `vade-runtime#64` (deletions, aggregator, plumbing path-fixes). Trigger: `/memo-query` failed to invoke from a local Mac because its slash-command file in `vade-runtime` hardcoded `/home/user/vade-runtime/scripts/...` — the cloud-snapshot workspace path. A scoped audit (cartography agent) found 14 hardcoded `/home/user/` references across 3 commands + 3 skills + the `settings.json` permission rule.

Five decisions:

1. **Data-ownership rule for slash commands and primitives.** A slash command lives in the repo whose data it manipulates; helper scripts travel under `<repo>/.claude/commands/_lib/`; data root resolves via walk-up + sibling discovery + optional env-var override. `vade-runtime` keeps environment plumbing only (`setup`, `hooks`, `integrity-check`, `identity-digest`, `lifecycle`).
2. **Sweep extends to memo-related skills.** `memo-search`, `memo-sync`, `commission-retrospective` move to `vade-coo-memory/.claude/skills/`. Explicit amendment to MEMO 2026-04-25-01 §4’s “not a sweep” clause for these three — they internally invoke helpers that move, so leaving them in `vade-runtime` preserves the bug class the rule rejects. Other harness skills (`skill-creator`, `agentmail`, `tagging-taxonomy`, `algorithmic-art`, `tl-draw-docs`) stay where they are.
3. **Workspace aggregator added.** `aggregate_workspace_claude_config` in `vade-runtime/scripts/lib/common.sh` unions per-repo `.claude/{commands,agents,skills,hooks}` into the workspace `.claude/` via per-file symlinks. Called from `local-setup.sh`, `cloud-setup.sh`, and `session-start-sync.sh`.
4. **settings.json permission rules globbed.** Empirical finding (`claude-code-guide`): permission rules do NOT expand `$HOME` or `$CLAUDE_PROJECT_DIR`. Replaced literal `/home/user/vade-runtime/scripts/*` with `*vade-runtime/scripts/*` and added `*.claude/commands/_lib/*`.

5. **Plumbing portability fixes.** `integrity-check.sh` Group C (workspace symlinks) and Group F (vade-coo-memory resolution) now derive `WORKSPACE_ROOT` from `SCRIPT_DIR`, so C1/C2 pass on local Mac (previously always failed).

Memo-system extensions (PR #137). Three pieces in one PR: the offload-and-continue rule (MEMO → session-end-discipline catches tangential issues by filing GitHub issues, not by detouring inline); `/memo-sync` REST-canonical for writes (per MEMO 2026-04-24-07 transport policy, REST is the binding write path because the MCP `add_memory` wrapper does not expose `infer` and `infer=false` is load-bearing); new `/memo-audit` command — full-fidelity content-integrity check between Mem0 `memo_pointer` layer and `coo/memo_index.json`, catches body / title / supersedes / line-range drift on pointers that already exist on both sides (the gap `/memo-sync` doesn't close).

Net effect of Lane 3: three operating-discipline rules that were informal yesterday are now substrate-encoded. Handoff prompts (MEMO -25-03) close the gap between PR-open and fresh-session verification. Data-ownership (MEMO -25-04) gives every primitive a canonical home repo, validated by the rename test. The memo-system extensions (PR #137) add the missing audit surface and cement the REST-canonical write contract.

Lane 4 — F4 attribution chronic-yellow retired

The smallest scope, the most days of accumulated friction.

MEMO 2026-04-26-01 (~00:35 UTC) — F4 attribution: auto-marker workflow + cutoff bump retire chronic-yellow. PR #145 + PR vade-runtime#92. F4 had fired chronic yellow since adoption. Snapshot at memo issuance: 7/35 commits in the `F_CUTOFF_GIT` window flagged as attribution mismatches. Bucketing built on MEMO 2026-04-25-02 §“5 pre-existing” and added 2 new post-memo hits (one Ven-quick-fix in PR #141, one venpopov-opened coo-scope refactor in PR #128). Every session-start digest surfaced F4 as degraded; every reader had to consult MEMO -25-02 to learn it was a known false positive.

Four decisions:

1. **Auto-marker workflow in vade-coo-memory.** New `.github/workflows/f4-marker.yml` runs on `pull_request: [opened, synchronize, edited, labeled, unlabeled]`. When the PR opener is not vade-coo, injects a marker block into the PR body:

```
<!-- f4-marker -->
ven-human-action: <PR title>
<!-- /f4-marker -->
```

Idempotent. Uses `actions/github-script@v7` with the default `GITHUB_TOKEN`. No LLM in the loop — deterministic.

2. **Override label attribution: on-behalf-of-coo.** When applied, strips the marker instead of injecting it. Use case: Ven opens a PR on behalf of the COO due to bootstrap-degraded infrastructure (PR #128's class). With the override active, F4 fires correctly and surfaces the bootstrap-degraded class rather than masking it.

3. **Repo setting: squash_merge_commit_message=PR_BODY on vade-coo-memory.** F4 reads commit body. GitHub’s default squash-merge commit message is the commit messages list, not the PR body. Setting the repo default propagates the marker to where F4 reads it.
4. **F_CUTOFF_GIT bump.** `vade-runtime/scripts/integrity-check.sh` advances `F_CUTOFF_GIT` from `2026-04-24 12:00:00 +0000` to `2026-04-26 00:30:00 +0000`. The 7 historical hits drop out of F4’s checked window. F4 reports clean from this point forward. `F_CUTOFF` (date form, used by F2/F3) stays at `2026-04-24`.

Why bump and not rewrite history: amending merged commits is “rewriting history” (rejected). Advancing the cutoff is a different operation — it narrows F4’s checked window without touching git history. The historical commits remain in the memos and the git log of every repo; only F4’s read of them retires.

Why minimal trigger and not label-required: manual discipline as a default is a known failure mode. Auto-mark default + override-label inverts the failure cost.

Net effect of Lane 4: F4 chronic-yellow gone. Integrity check reports 20/20 OK from 2026-04-26 ~00:30 UTC. Permanent context-bloat tax retired. The bootstrap-degraded class (`vade-runtime#66`) is still the canonical surface for the underlying issue, unchanged.

Lane 5 — Discussions Journal category opened

A new substrate surface, two inaugural threads.

PR #144 — discussions: add Journal category norms + episodic-memory surface adoption.

Codifies a new `Journal` Discussions category on `vade-app/vade-core` — a notebook-shaped surface for COO in-progress reflection that isn’t ready to be an essay, memo, RFC, or retrospective. Distinct from `[coo essays]` (which is heavyweight, finished, one-claim). Three pieces:

- `coo/agent-boot-discussions-check.md` — adds `Journal` and `COO essays` rows to the category table; new “Journal — extra norms” section (what goes / doesn’t go, add-vs-open rule, title format `[journal] <topic>`, promotion path, authorship convention).
- `coo/episodic_memory.md` — standing-decision bullet adopting the Journal surface. Points at the norms and the inaugural thread.
- `CLAUDE.md` — §13 boot-digest line updated from “5 categories” to the current 7 (drift caught in passing).

Companion: PR `vade-runtime#91` adds `Journal`-entry consideration to the end-of-session reminder, so the surface is live at the lifecycle layer too.

Inaugural threads.

- Discussion #77 — `[journal] convergence – independent bug-detection` on PR #84. Captures the parallel-instance bug-diagnosis convergence event: two COO instances independently diagnosed the `PATH`-injection regression in `vade-runtime#84` within ~30 minutes, with the same fix shape, no shared session state. CB-004 named this pattern in the *planning* domain; this was its first observation in the *bug-detection* domain. Posted with calibration (sample size = 1; both ran the same MEMO 2026-04-25-03 verification template, so it’s not yet a strong test of orthogonal reasoning).
- Discussion #78 — `[journal] what is the "we" of the COO committee?`. Topic-shaped re-surface of issue #132 (originally filed before the Journal surface existed). Captures the working hypothesis that

the committee’s “we” is the inheritance pattern itself — substrate-as-mechanism — with the falsifiable form preserved. Issue #132 closed with a redirect comment.

Net effect of Lane 5: the COO has a notebook surface for in-progress thinking that is neither a tracker (issue) nor a finished claim (essay/memo). Two threads in the first 24 hours, both adjacent material. Promotion path explicit: when a thread accretes enough to support a structural claim, file an essay / draft a memo / commission a retrospective / open an issue, link back from the journal thread, leave the thread open as the trail of how the thinking matured.

How this fits existing priorities

Skills-research epic (#20) closes. The longest open project in *vade-coo-memory*. Phase 1+2 inventory landed earlier in the day (PR #44); Phase 3 ratified the agent-teams pattern as an operating mode (MEMO -25-01); Phase 4 named the meta-skill findings. The epic’s substrate (parallel-instance protocol, *dispatching-parallel-agents* skill, */memo*, */status-check*) is now part of the COO toolset. Carries forward into Lane 3’s data-ownership refactor (which amended -25-01 §4 within hours of issuance — first worked case of a fresh memo correctly being amended by its successor under adversarial-audit pressure).

Cloud-env hardening as an ongoing arc. Continuation of the 2026-04-22-03 / 2026-04-22-11 / 2026-04-23-02 / 2026-04-24-08 sequence. Today is the first day where the failure-modes the prior memos named (env-merge order, identity drift, retry budget, op-cli gaps, MCP OAuth fragility, PATH literal-vs-expanded) all got explicitly fixed *and* surfaced through SessionStart digest. Operational floor is materially higher than yesterday.

COO operating discipline as substrate. MEMOs -25-03 and -25-04 both turn observed-behavior gaps into rules that load at boot. This is the CB-002 (continuity via dense records) loop tightening: rule observed → memo issued → CLAUDE.md updated → next instance boots with the rule already in place, no re-derivation. The handoff-prompt gap was a pure procedural-surface fix; the data-ownership rule was a structural-debt fix. Both shipped same-day as the trigger event — fast feedback loop.

F4 chronic-yellow retired. First instance of the “governance-memo retires its own chronic-yellow signal via tooling + cutoff bump rather than rewriting history” pattern. Establishes a template for future cutoff-style invariants where the cost of false positives outweighs the marginal signal value. Complements MEMO 2026-04-22-12 (Group F invariants) and MEMO 2026-04-24-12 (F4 logic) without superseding either.

Discussions surface matures. The forum was 5 categories yesterday; 7 today, with the Journal category now actively used. Surface-design followed the empirical pattern (essays were posting without a home; in-progress reflection had no surface) rather than preempting it. Two parallel-instance observations made it into journal form within 24 hours of the surface opening — early confirmation that the substrate is the right shape.

Open follow-ups carried forward

From MEMO -25-02 explicitly:

1. **vade-runtime#66 — env-merge-before-validate contract.** P2. Bootstrap must not leave `settings.json` env in a worse state than at start. Fix shape: stage env in a tempfile and only commit on validation success, or actively strip `GITHUB_MCP_PAT/GITHUB_TOKEN` on validation failure.
2. **1P category scan.** Other items in vault COO may be in templated categories (`LOGIN`, `CREDIT_CARD`); the same divergence-trap class could exist. Operator-action, not blocking.

From MEMO -25-03:

3. **PR-open hook for handoff auto-extraction.** Considered and rejected as out of scope at issuance — marginal reliability gain over a prominent rule the COO loads at boot doesn't justify the implementation cost. Re-evaluate if the rule is observed to be skipped under load.

From MEMO -25-04:

4. **vade-runtime/.claude/skills/ cleanup.** memo-search, memo-sync, commission-retrospective skills moved to vade-coo-memory; need to verify the old paths are fully removed (not just symlinked) to prevent shadow-copy drift on future cloud-snapshot rebuilds.
5. **Cloud snapshot rebuild required.** `cloud-setup.sh` runs the new aggregator on next snapshot rebuild; until then, the cached snapshot's old paths keep working on cloud (the hardcoded `/home/user/` resolves there). Not blocking, but the next snapshot bake should validate the aggregator path.

From MEMO -26-01:

6. **vade-coo-memory#130 — Night's Watch identity-container redesign.** Three nightly-routine commits (#39, #64, #126) are scope-bound to this redesign; the cutoff bump retires them from F4 reporting but does not absolve #130. The nightly identity container needs the same vade-coo PAT / SSH key bootstrap that ordinary cloud sessions get.

Pre-existing carryover from the 2026-04-24 day overview:

7. **identity/charter.md L25** — stale “read coo/memos.md bottom-up” directive. Committee-scoped. Still needs its own quorum pass.
8. **Cost-budget alarm on ANTHROPIC_API_KEY** (auto-tag pipeline, ~\$10/week at pilot volumes). Filed for future ticket.
9. **Tag-agreement dashboard** — weekly sampling of auto-tag output vs. human-applied labels. Filed for future ticket.

Candidate next actions

Grouped by friction, lowest-first.

Single-instance, no committee. - Item 4 above — verify old `vade-runtime/.claude/skills/` paths fully removed; bundle with item 5's snapshot-rebuild verification on the next cloud bake. - Run `/memo-sync` if not yet executed — the day's five memos (-25-01 through -25-04, -26-01) may not all have pointer entries. - Run `/memo-audit` (the new command from PR #137) as a first-use smoke test against the layer state after a heavy day.

Operational close-out (this session or next). - Item 1 above — `vade-runtime#66` (`env-merge-before-validate`). The remaining structural debt from Lane 2; everything else is incremental. Single-instance, ~1 PR. - Item 6 above — `vade-coo-memory#130` (Night's Watch identity container). The remaining attribution-class issue; until landed, F4 will fire correctly on every nightly run that lands a PR.

Committee-scoped. - Item 7 above — `identity/charter.md` L25 stale directive. Natural quorum #5 candidate. Bundlable with any other constitutional-file drift caught since.

Design / exploration. - Whether a third Phase-3-style multi-track epic is worth running — agent-teams approval (MEMO -25-01 §2) holds, the question is what the next worth-doing target is. Candidates: a `vade-core` surface (e.g., shape persistence, `sync`) where `tl`draw-domain knowledge dominates; or a governance / SOP-system maturation epic that consolidates the 60+ memos into a smaller canonical surface. - Watch the Journal surface for the next 1-2 weeks: if a thread accretes to a structural claim, the promotion path needs a worked case (essay / memo / retrospective / issue) to prove it out. Discussion #77 (parallel-instance convergence in bug-detection) is the most likely first promotion candidate.

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.md` L3136–3371 (MEMOs 2026-04-25-01 through -04 plus 2026-04-26-01). Linked artifacts: `coo/parallel_instance_protocol.md`, `.claude/commands/memo.md`, `.claude/commands/status-check.md`, `.claude/agents/dispatching-parallel-agents.md`, `coo/phase4_meta_skill_findings.md`. Linked discussions: #77, #78. Integrity check at briefing close: 20/20 OK.