

Write Loops, Not Prompts: Cowork, Codex Sites, and Lakebed

Coding Agents Alpha Tracker

2026-06-07

Write Loops, Not Prompts: Cowork, Codex Sites, and Lakebed

By Coding Agents Alpha Tracker • June 7, 2026

Power users are moving from single-chat prompting to orchestration loops, while Claude Cowork, Codex Sites, Hermes Desktop, and Lakebed all make that power easier to deploy and easier for non-terminal users to access. This brief gives copyable workflows for onboarding, internal tools, shared backend skills, and agent-friendly deployment.

TOP SIGNAL

The bottleneck has moved *above* the model. Boris Cherny says his Claude Code workflow went from direct prompting to running 5-10 agents in parallel, and now to writing orchestration loops that do the prompting for him [1]. Greg Brockman describes the same overhang from the Codex side: when he does not use Codex, it is usually because context is missing, a skill has not been written, or he forgot to use it—not because the task exceeds model capability [2]. Theo’s Lakebed pitch is the infra version of that insight: agents are decent at writing code but bad at navigating dashboards, so the opportunity is giving them code-native primitives they already know how to use [3].

“My job is to write loops.” [1]

TRY THIS

- **Boris Cherny (Anthropic) — parallelize the agent, then abstract one layer up.** Start by opening multiple Claude Code sessions in parallel—Cherny says he was already running 5-10 at once—then move repeated coordination into your own loops or scripts so the agent does the prompting and task selection for you [1].

- **Use the agent as the repo onboarding surface.** At Anthropic, new engineers ramp in about two days by running Claude Code inside the codebase and asking it questions that used to require hunting through docs or internal tribal knowledge; Cherny’s example is database access itself: instead of explaining how to query it, have Claude do the query from repo context [1].
- **Riley Brown — turn a prompt into a team-only internal tool, then add a shared backend skill.** In Codex, use the Sites plugin and connected apps like Gmail and Slack, then try:

“Please build a dashboard of the most important emails and Slack updates from this week.” [4]

Brown’s follow-on pattern: store the tool’s data in Convex, create a /Kanaban skill, and let *any* chat thread—or even another agent on another platform—append items to the same board [4].

- **Theo — hide cloud plumbing behind agent-friendly commands.** Lakebed’s demo flow is unusually copyable:

1. Run `npm lakebed new` to create a project [3]
2. Run `npm lakebed deploy` to get a live app with database, auth, and sync [3]
3. In Cursor Composer Fast, prompt:

“Make this app a real kanban with a live chat that shows users names when done deploy with NPX lakebed deploy” [3]

The agent edits the app and updates the live deployment in place [3]. Theo’s broader lesson: agents are better with TypeScript/Vite-style primitives than dashboard clicking [3].

WHAT SHIPPED

- **Claude Cowork** — Boris Cherny says Anthropic shipped a non-terminal surface for Claude Code; the team built it in roughly 8-9 days and 100% with Claude Code itself, aimed at users who want Claude Code’s power without terminal setup [1].
- **Codex Sites + build iOS apps plugin** — Riley Brown shows Sites deploying generated apps as internal, team-only web tools with connected data, while the iOS plugin renders generated Swift apps live in the Codex sidebar/browser [4].
- **Lakebed (in progress)** — Theo demoed an agent-first cloud layer where agents build and deploy full apps through code rather than dashboards or bespoke config; today’s signal is the live `npm` scaffold-to-deploy workflow [3].
- **Hermes Desktop** — Riley Brown describes Hermes moving from a terminal-first install to a desktop app with agents, skills, messag-

ing/channel connections, artifacts/files, and model switching across Anthropic, OpenAI, DeepSeek, and others [4].

- **Model/adoption signal: DeepSeek v4 Pro** — Brown cites Lindy.ai switching 100% of traffic for agentic workloads to DeepSeek v4 Pro, with reported gains on core tasks at \$1.3/M tokens—23x cheaper than Opus 4.8 and almost 27x cheaper than GPT 5.5 [4].
- **Anthropic model signal** — Cherny says the biggest Claude Code jumps lined up with Sonnet 4 / Opus 4 and later Opus 4.5; Anthropic’s last public figure was about 3x more code per engineer, and he says it is higher now [1].
- **Codex app vs CLI is still split** — scaling01 says five Codex CLI tabs beat the app for parallel multi-project work, while Theo says the app wins because threads stay sorted by project [5, 6].

GO DEEPER

- **11:35-11:53** — **Boris Cherny on the next abstraction layer.** Short but high-signal: multiple Claude Code sessions in parallel, then one more layer up where the human stops prompting and starts writing loops [1].



[] : Claude Code / WorkOS Acquired Unplugged (11:35)

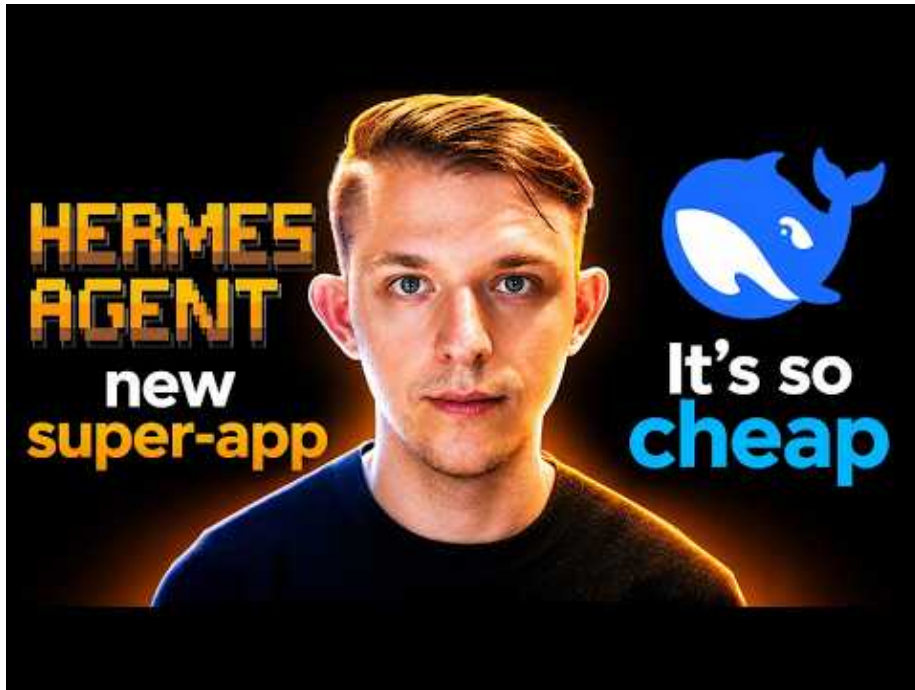
- **23:28-24:43** — **Theo’s Lakebed demo from blank project to live kanban.** Watch the exact command-and-prompt flow that turns an npx scaffold into a deployed collaborative app, then ask whether your infra is

equally usable by an agent [3].



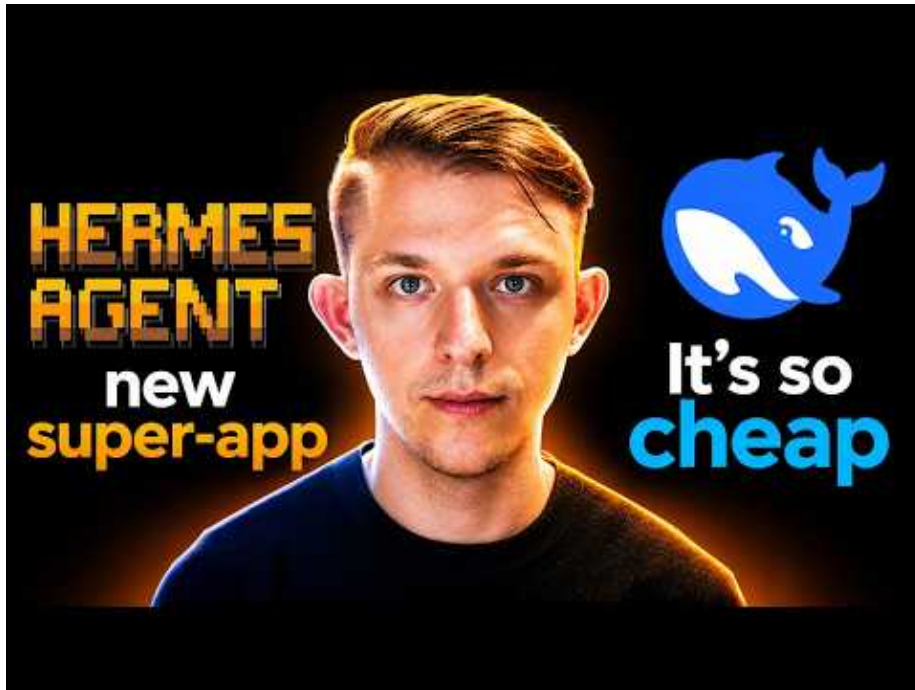
Vite is now part of Cloudflare (23:28)

- **2:32-4:02** — **Riley Brown's /Kanaban pattern.** Good example of a reusable agent skill sitting in front of a shared backend: once the skill exists, any chat or agent can update the same board without reopening the original app [4].



HUGE Cursor Update and DeepSeek v4 Catches Up To Opus 4.8? (2:32)

- **0:38-1:49** — **Codex Sites as internal-tool generator.** Useful if you want the simplest mental model for the new Sites feature: connect company data, write one prompt, and get a team-only web app back [4].



HUGE Cursor Update and DeepSeek v4 Catches Up To Opus 4.8? (0:38)

Editorial take: the highest leverage right now is packaging model capability into loops, skills, and deploy surfaces—not asking the same chat box for longer answers.

Sources

1. [] : Claude Code | WorkOS Acquired Unplugged
2. X post by @gdb
3. Vite is now part of Cloudflare
4. HUGE Cursor Update and DeepSeek v4 Catches Up To Opus 4.8?
5. X post by @scaling01
6. X post by @theo