Codex Orchestration Patterns Solidify as Opus 4.7 Token Costs Surface

Coding Agents Alpha Tracker

2026-04-20

Codex Orchestration Patterns Solidify as Opus 4.7 Token Costs Surface

By Coding Agents Alpha Tracker • April 20, 2026

Riley Brown's end-to-end Codex walkthrough and Alexander Embiricos' subagent note converged on the same playbook: long-lived threads, steering, forks, and custom skills. Simon Willison added the cost reality check, quantifying how Claude Opus 4.7 can materially inflate context budgets.

TOP SIGNAL

Riley Brown's Codex walkthrough was the clearest practitioner signal today: the winning pattern is no longer *prompt once, wait*, but *run multiple long-lived threads, steer them mid-flight, fork when a branch deserves its own context, and turn repeat tasks into custom skills* [1]. Alexander Embiricos described the same move in miniature: keep a thread active, then use a subagent in parallel when new work arrives [2]. The durable takeaway is that coding-agent leverage is shifting from single prompts to orchestration [1, 2].

TOOLS & MODELS

- **Codex desktop + GPT-5.4 (extra high):** Riley's default setup is full access with GPT-5.4 on extra high effort. The practical differentiators are project folders, parallel chats, steering, fork-into-local, and unified browser/computer control [1].
- **Codex vs Claude Code:** Riley's current split is straightforward: Codex has the better interface and multitasking model, while Claude is better for design-heavy work, so he routes those tasks accordingly [1, 3].
- **Claude Opus 4.7 vs 4.6:** Simon Willison measured 7,335 tokens for the same system prompt on Opus 4.7 vs 5,039 on Opus 4.6 — **1.46x** more

tokens. At the same $5/M input and $25/M output pricing, that implies roughly **40% higher cost** on that kind of workload [4].
- **Vision token inflation is even bigger:** Simon's image test came in at 4,744 tokens on Opus 4.7 vs 1,578 on Opus 4.6 — **3.01x** more tokens — though 4.7 also supports images up to **2,576px** on the long edge [4].
- **Useful utility:** Claude Token Counter now compares Opus 4.7, Opus 4.6, Sonnet 4.6, and Haiku 4.5 in one UI. If you care about system-prompt size or huge contexts, use it before swapping models blindly [4].

WORKFLOWS & TRICKS

- **Serial-task instead of waiting.** Riley's rule: put real effort into the prompt, press enter, then move to the next chat. Embiricos' variant is to keep longer-lived threads warm and call a subagent in parallel for new work [1, 2].
- **Package repeat work as a skill.** Riley's loop is replicable: **(1)** identify the annoying repeated task, **(2)** find the API, **(3)** run `/skill creator`, **(4)** paste docs/API key, **(5)** let Codex generate the skill, **(6)** open a fresh chat to use it, **(7)** automate it once it proves useful [1].
- **Steer live; fork cleanly.** Don't save feedback for the next run. Riley pastes screenshots and uses **Steer** to inject corrections mid-task; when a branch becomes a separate deliverable, he forks the chat into local and renames it as a new workstream [1].
- **Full-stack bootstrap inside one agent workspace.** His `Chorus` flow was: create a Swift hello-world app and let Codex open Xcode/simulator, generate screens with a custom mobile-design skill, integrate those files, add Supabase Postgres via MCP, switch auth to simple email/password, scaffold a React landing page with a Tally embed, deploy to Vercel, then prep TestFlight/App Store [1].
- **Trust, but verify every external integration.** Riley asked for Typefully **V3**, but later found the generated control path was **V2**; he checked that the draft was actually created before planning automations. Same lesson applies to any agent-wired API: verify side effects before you scale them [1].
- **MCP gotcha:** after adding Supabase remote MCP support, Riley had to restart Codex before the session could see the new server and apply the DB changes [1].

PEOPLE TO WATCH

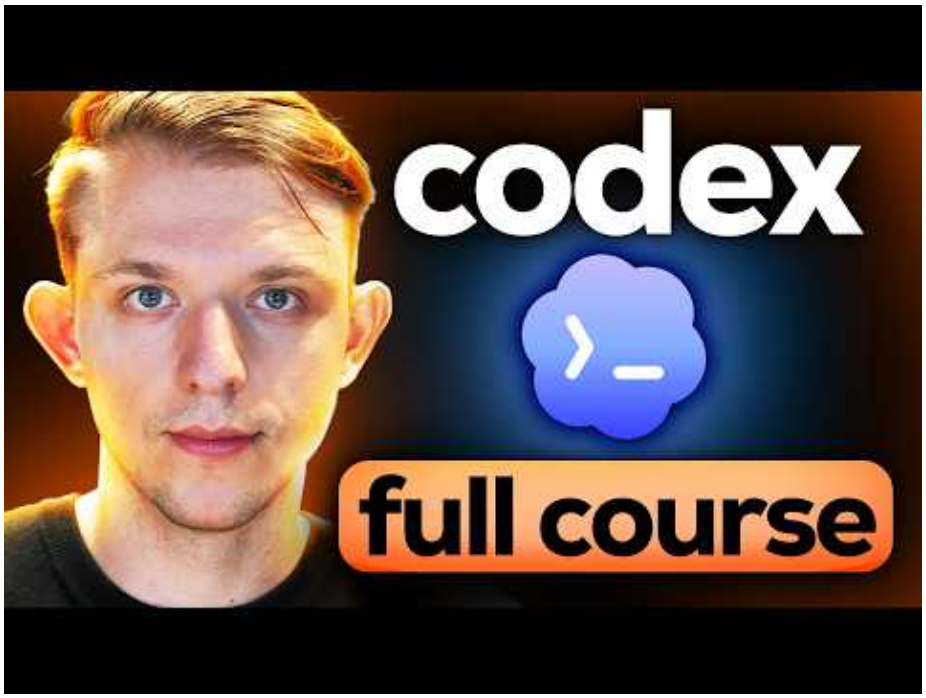
- **Riley Brown** — Most practical Codex content drop of the day. He goes past interface demos and shows a real build path: Swift app, Supabase, Tally, Vercel, device testing, and TestFlight [1].
- **Alexander Embiricos** — High-signal because the advice is short and operational: longer-lived threads, automation-pinged context, and subagents in parallel [2].

"Subagents + steering in Codex is pretty magical." [2]

- **Simon Willison** — Best model-cost reality check today. He turned tokenizer changes in Claude Opus 4.7 into concrete numbers you can use for prompt and budget planning [4].
- **steipete / swyx** — If you maintain agent tooling or accept third-party skills, watch this thread. swyx's recap of steipete's OpenClaw update includes some of the clearest operational security numbers in the space: **60x** more security reports than curl and an estimate that **12%-20%** of skill contributions are malicious [5].

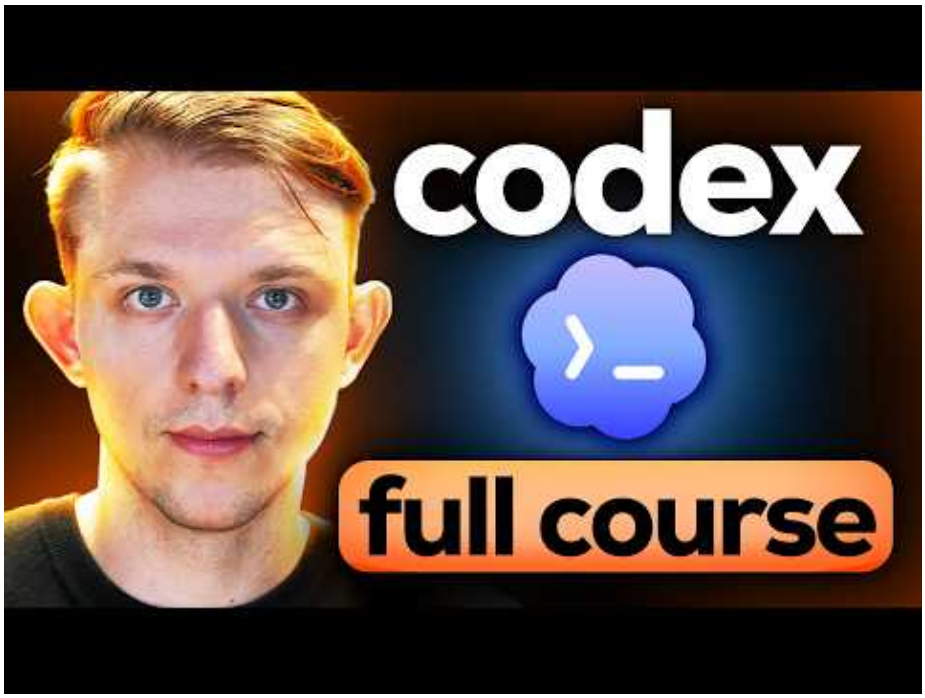
WATCH & LISTEN

- **28:17-32:24 — Turn an annoying manual task into a reusable skill.** Riley walks through the whole loop: find an API, call `/skill creator`, paste the key, let Codex build the tool, then reopen in a new session to use it. This is the cleanest clip here if you're still re-prompting instead of packaging repeat work [1].



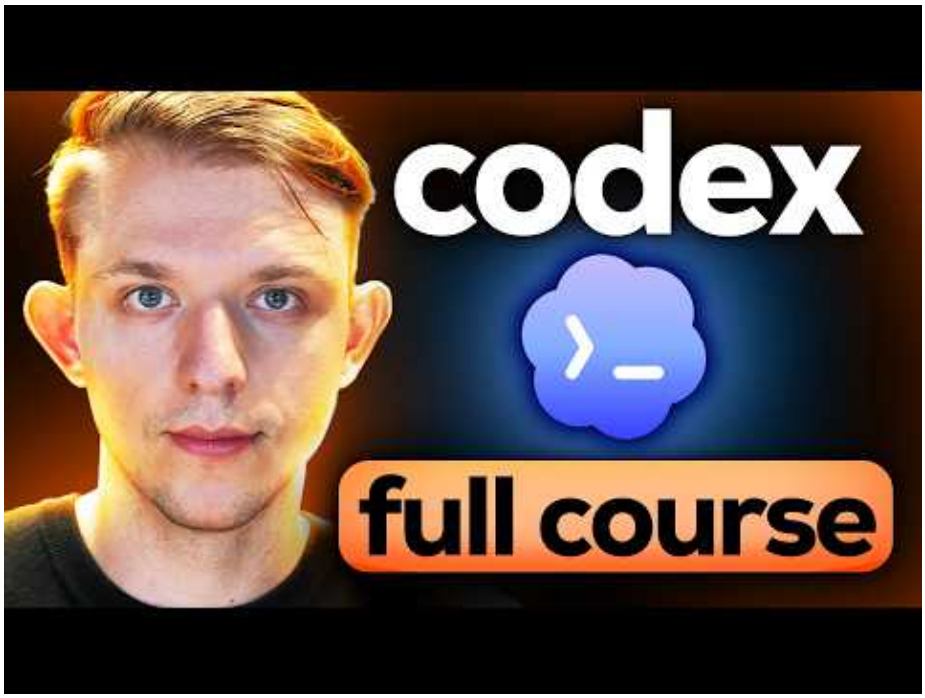
Codex Full Course 2026: The NEW Best AI Coding Tool (28:17)

- **46:11-48:40 — Ship the waitlist before the polish.** He scaffolds a React landing page, embeds a Tally form, and gets the page running locally fast. Good pattern for agent-built products that need demand capture before design perfection [1].



Codex Full Course 2026: The NEW Best AI Coding Tool (46:11)

- **1:02:13-1:04:27 — Real MCP workflow, warts included.** Riley hits a live limitation, restarts Codex so the new Supabase MCP server is visible, then verifies the generated tables and app data. Boring clip, useful clip [1].



Codex Full Course 2026: The NEW Best AI Coding Tool (62:13)

PROJECTS & REPOS

- **Remodex** — Open-source Codex remote control for iOS. One QR scan pairs your phone with Codex on Mac; from there you can create threads, use subagents and skills, run Git actions, and keep the connection E2EE. It is already live on the App Store, and Riley says his own simpler internal remote borrowed from the repo [6, 7, 8].
- **OpenClaw** — Not a shiny feature drop; a serious security signal. swyx's recap of steipete's five-month update says the project is already dealing with nation-state attacks, a flood of security reports, and a nontrivial percentage of malicious skill submissions — useful data if you're building any agent ecosystem with user-contributed tools [5].

Editorial take: the highest-alpha skill right now is agent orchestration — threads, subagents, reusable skills, and cost-aware model routing matter more than any single flashy completion [1, 2].

Sources

1. Codex Full Course 2026: The NEW Best AI Coding Tool
2. X post by @embirico

3. X post by @rileybrown
4. Claude Token Counter, now with model comparisons
5. X post by @aiDotEngineer
6. X post by @emanueledpt
7. X post by @rileybrown
8. X post by @rileybrown