Secure Compute and Model Routing for Long-Running Agents

Coding Agents Alpha Tracker

2026-07-24

Secure Compute and Model Routing for Long-Running Agents

By Coding Agents Alpha Tracker • July 24, 2026

Today's strongest signal is the move from one-shot coding assistants to long-running, isolated agent systems. Practical takeaways cover disposable sandboxes, phase-based model routing, Slack-to-PR loops, and the emerging UX of managing agent work as an inbox.

TOP SIGNAL

Long-running coding agents are becoming an infrastructure problem, not a prompting problem. LangChain says agent tasks it sees now take about 20 minutes, versus one to five minutes a few years ago; its answer is a separate disposable computer per agent, with snapshot/fork support and isolation that can scale to thousands of concurrent sandboxes. [1] Codex's `/goal` points in the same direction: assign an ambitious task such as a large refactor, then let the agent run uninterrupted for hours or days. [2]

TRY THIS

- **Give each unattended coding task its own disposable VM.** Mount the repository from GitHub, GCS, or S3; optionally start from an image containing your usual dependencies; then snapshot the environment before risky changes so you can restore or fork competing approaches. LangSmith Sandboxes supports these mounts, bring-your-own images, snapshots, and isolated hardware-virtualized micro-VMs. [1]

  For untrusted model-generated code, keep credentials out of the runtime and route outbound access through a proxy that controls egress. That operationalizes Kent C. Dodds's advice to stop manually shuffling context and instead give the agent *secure* access to the services it needs. [1, 3]

- **Route models by phase, not by habit.** Matthew Berman's personal workflow: (1) give the hardest task to Fable to inspect the codebase and write a full spec; (2) hand that plan to a cheaper execution model such as Grok 4.5 or Cursor Composer; (3) give the finished diff and spec to GPT-5.6 for an independent review. [4] In his reported same-task comparison, the mixed workflow cost $25.55 versus $81 with Fable alone and $46.50 with GPT-5.6 alone. [4]
- **Turn a Slack request into a reviewed PR loop.** Trigger an agent from a message such as: *"Update the docs to showcase verification loops on a new goal or rubric page."* Have it draft the PR, run a verifier for requirements such as resolved links and passing CI, then post the PR back to Slack for human review and merge. LangChain uses this shape for its docs agent. [5]
- **Treat agent threads as an inbox.** In T3 Code's nightly sidebar, settle a finished thread to move it to the bottom; merged PRs auto-settle their related threads, while messaging a settled thread reactivates it. Theo says this creates an incentive to land PRs and clear the sidebar. [6, 7, 8]

WHAT SHIPPED

- **LangSmith Sandboxes:** isolated micro-VMs for agents with roughly one-second median spin-up. LangChain positions them for agents that write, execute, and test code without provisioning local environments; every line executed by its Open Suite demo runs inside the sandbox. A free allocation of five LCUs and one LSU—about 100 minutes of sandbox use—was announced for all plans. [1]
- **Voice control for ChatGPT Work and Codex:** OpenAI says desktop-app users on macOS and Windows can control the computer and direct multiple agents by voice. It is powered by GPT-Live, which can speak, listen, and coordinate work simultaneously, and is rolling out to Plus, Pro, Business, Edu, and Enterprise plans. [9]
- **ChatGPT Sites:** Simon Willison reports that ChatGPT's Work mode can build and deploy public websites on Cloudflare Workers with SQLite persistence—taking an idea directly to a hosted app. [10, 11]
- **T3 Code sidebar:** now available behind a settings toggle in the latest nightly build. Its opinionated thread lifecycle—working threads visually de-prioritized, settled threads sorted by settle time, merged PRs auto-settled—is a notable UX experiment for multi-thread agent work. [6, 7]
- **Tool-calling regression watch:** Peter Steinberger says newer Claude Opus/Sonnet versions caused tool-invocation failures with Pi's edit tool where older versions worked; his team added code paths that call the Claude CLI directly as a workaround. The linked writeup is a reminder to regression-test the harness, not just benchmark model quality. [12, 13]

GO DEEPER

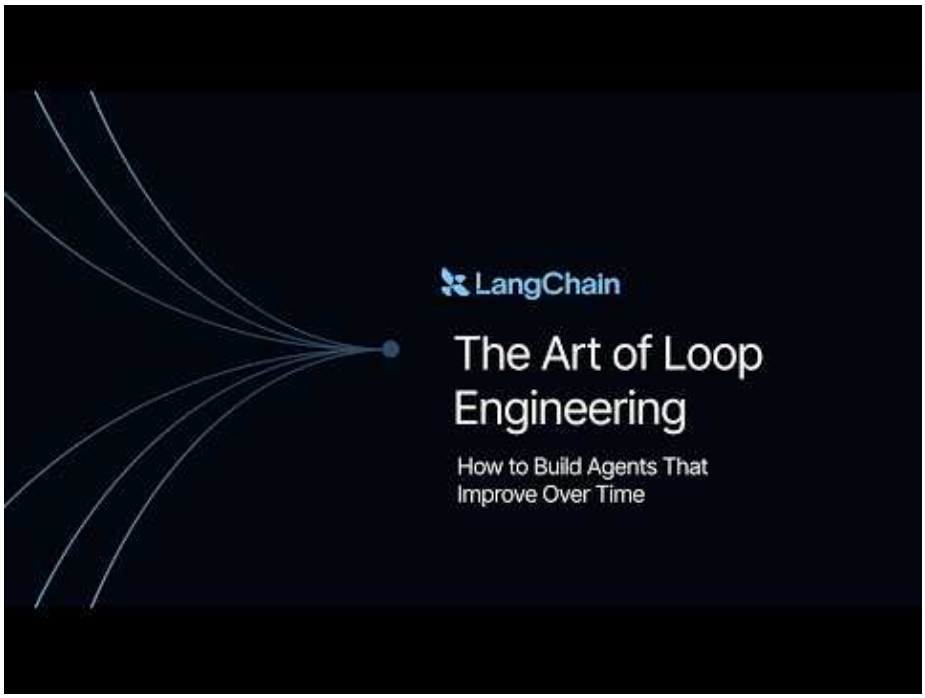
- **11:55–12:52 — Why containers are not enough for agents.** LangChain explains the specific threat model: agents may install arbitrary dependencies and execute model-generated scripts, while an auth proxy keeps credentials outside the sandbox runtime and controls egress.

Build a secure computer for your agent (11:54)

- **8:14–9:32 — A concrete planner → executor → reviewer routing stack.** Matthew Berman walks through using a high-capability model for the spec, a cheaper model for output-heavy implementation, and GPT-5.6 to audit the result against the original requirements.

You're probably wasting tokens (8:14)

- **10:16–13:33 — Event-driven docs agent with verification and review.** Watch the full Slack-triggered loop: request, PR draft, link/CI verification, Slack notification, and human merge. It is a compact template for putting agents where work already happens.



The Art of Loop Engineering: How to Build Agents That Improve Over Time (10:16)

- **Repo: jediahkatz/ai-proofs.** Jediah Katz released the materials after reporting that Cursor helped disprove a longstanding conjecture about a modified randomized greedy algorithm. His reported setup used Fable as the main driver, with Sol and Grok as adversarial reviewers—without elaborate prompt engineering. [14, 15]

Editorial take: as agents run longer and touch more systems, the durable advantage is a controlled execution environment, phase-specific model routing, and a review loop that produces evidence—not a bigger prompt. [1, 4, 5]

Sources

1. Build a secure computer for your agent
2. Romain Huet (OpenAI) on Codex & GPT-5.6, my AI papercut spree & Insecure Agents crossover
3. X post by @kentcdodds
4. You're probably wasting tokens
5. The Art of Loop Engineering: How to Build Agents That Improve Over Time
6. X post by @theo
7. X post by @theo

8. X post by @theo
9. X post by @OpenAI
10. X post by @simonw
11. X post by @reach_vb
12. X post by @mitsuhiko
13. X post by @steipete
14. X post by @jediahkatz
15. X post by @jediahkatz