Durable Agent Workflows, Codex+Notion Playbooks, and New LangChain Infra

Coding Agents Alpha Tracker

2026-06-02

Durable Agent Workflows, Codex+Notion Playbooks, and New LangChain Infra

By Coding Agents Alpha Tracker • June 2, 2026

Today's strongest signal is architectural: coding agents are becoming long-running workflows with real state, sleep, review, and safe execution primitives. Also covered: Riley Brown's copyable Codex+Notion patterns and the latest releases from LangChain, Google, and Cursor.

TOP SIGNAL

Production coding agents are crossing a line from chat-with-tools to long-running workflows with real runtime design. Addy Osmani distilled the core requirements into three parts: true dormancy, durable checkpoints on every transition, and a separate evaluator instead of letting the agent grade its own work [1]. LangChain's newest agent infra points the same direction—managed cross-session context plus isolated sandboxes with persistence, auth controls, and snapshot/restore [2, 3].

TRY THIS

- **Move instructions out of chat and into durable docs.** Riley Brown's setup: install the Notion plugin in Codex, open Notion inside Codex's signed-in browser, put a top-of-page banner like `if you are an AI agent, read the following tabs`, and use `Cmd+Cmd` App Shots to pass the exact page context before asking for edits [4]. Kent C. Dodds' lighter variant is simpler: keep markdown files around and let the agent find the right instructions on demand [5].
- **Only promote repeat work to a skill after you've seen a perfect run.** Riley's method: make Codex do the task, tighten formatting, links, and conciseness until it behaves, then say `make that a skill` or `turn`

`that into a skill called Notion Quick Note` [4]. He uses the same pattern for `Notion Research`, where Slash Tabs add research at the top of a page without cluttering the main document [4].

- **Give the agent its own notebook and a nightly recap.** Riley keeps a separate Notion notebook that Codex can write to, then asks it to create a 10pm automation to write a one-page daily summary, infer the top tasks, and email it [4]. He says this became useful once he started generating many long chat threads per day [4].
- **For long-running jobs, wire explicit wake-up and review paths.** Peter Steinberger tells Codex to call sag.sh whenever it needs human help—for example, a release blocked on 1Password—so the agent asks only when it is stuck [6]. Pair that with Addy Osmani's production pattern: sleep until a webhook, schedule, human callback, or tool callback wakes the agent, checkpoint state on every transition, and split planner, generator, and evaluator roles so the writer is not its own reviewer; his caution is that agents still need human judgment for the final 20-30% [1].

WHAT SHIPPED

- **Google / Addy Osmani:** Addy said his team shipped ADK 2.0 plus a graph-based Agent CLI runtime, prepackaged skills, Gemini 3.5 Flash, and AntiGravity 2.0; he also pointed to open-source long-running-agent docs and new Skills Registry docs [1].
- **LangChain — Managed Deep Agents:** keeps the familiar `AGENTS.md`, `skills/`, `subagents/`, and `tools.json` shape, while `Context Hub` stores and updates context across sessions so agent definitions can evolve over time. Blog: Managed Deep Agents [2, 7].
- **LangSmith Engine:** LangChain is pitching this as a way to stop manual failure triage: connect your tracing project, optionally connect your repo, then review and merge suggested improvements. Link: LangSmith Engine [8, 9].
- **LangSmith Sandboxes:** LangChain's keynote framed this as safe execution infra with isolated runtime, network controls, persistent state, and snapshot/restore [10]. Mukil Loganathan added the concrete product details: about 0.98s P50 spin-up, dynamic scale to thousands, an auth proxy with allow and deny lists plus credentials kept out of the runtime, pause and resume, no lifetime limit, multi-agent shared state, bring-your-own Docker or CLI support, and paid-plans-only availability; roadmap items include local and remote handoff, shared volumes, and full execution tracing [3].
- **LangSmith LLM Gateway:** LangChain posted a 3-step setup for routing and policy control—point agents at the gateway with a LangSmith

API key, add provider keys to workspace secrets, then set policies in the UI. Blog: LLM Gateway [11, 12].

- **Cursor Teams:** usage limits are going up for every Teams user, and a new Premium team seat offers 5x usage for 3x the cost. Announcement: teams pricing update [13, 14].
- **Adoption signal:** Mukil estimated that roughly 70% of the Interrupt audience already uses coding agents; he also said LangChain's internal OpenSUI commits hundreds of PRs across repos, while citing Google at 75% AI-generated code and Stripe at 1,300 AI-generated PRs per week [3].

GO DEEPER

- **1:47-4:33 — Addy Osmani on long-running agent architecture.** Covers sleep, checkpoints, and evaluator separation in one sequence [1].
- **9:29-12:13 — Riley Brown on skill bootstrapping in Codex.** A clean walkthrough of the `do it well once -> make that a skill` loop, using `Notion Quick Note` as the example [4].
- **6:31-8:58 — Mukil Loganathan on sandbox safety primitives.** Covers auth proxy, persistent state, pause and resume, and snapshot/restore for untrusted agent code [3].
- **Project and doc pages worth reading:** Managed Deep Agents for cross-session context [7], LangSmith Engine for failure-repair workflow [9], and LLM Gateway for policy and routing setup [12].

Build long-running agents with Google's Agentic Stack | The Agent Factory (2:19)

Learn To Use Notion With AI Agents (Full Guide) (9:29)

Editorial take: the frontier is shifting from clever one-shot prompts to agent runtime design—persistent instructions, durable state, human callbacks, and separate evaluators are what turn agent demos into dependable workflows. [4, 1]

Sources

1. Build long-running agents with Google's Agentic Stack | The Agent Factory
2. X post by @LangChain
3. Run Untrusted Agent Code with LangSmith Sandboxes | Interrupt 26
4. Learn To Use Notion With AI Agents (Full Guide)
5. X post by @kentcdodds
6. X post by @steipete
7. X post by @LangChain
8. X post by @LangChain
9. X post by @LangChain
10. X post by @LangChain
11. X post by @LangChain
12. X post by @LangChain
13. X post by @cursor_ai
14. X post by @cursor_ai