Harnesses Become the Real Lever as Codex Lands in Claude Code

Coding Agents Alpha Tracker

2026-03-31

Harnesses Become the Real Lever as Codex Lands in Claude Code

By Coding Agents Alpha Tracker • March 31, 2026

The best signal today is that coding-agent performance is increasingly a harness problem, not just a base-model problem. Also inside: the open-source Codex bridge into Claude Code, practical workflows for local models and secure orchestration, and the clips worth watching.

TOP SIGNAL

The strongest signal today: **the harness is now a first-order performance variable**. Georgi Gerganov says most local coding-agent failures come from the harness, chat-template, prompt-construction, and inference chain—not just the model [1], Matt Maher's 100-feature PRD benchmark found **Cursor** improved frontier-model results by **11% on average**, with **Opus** scoring **20% higher** there than in Claude Code [2, 3], and the open-source **Meta Harness** paper summary says changing the harness around a fixed model can create a **6x** gap [4].

For builders, benchmarking the base model alone is increasingly the wrong abstraction; routing, review, retrieval, debugging visibility, and context handling are where a lot of the practical edge is moving [5, 6, 7].

TOOLS & MODELS

- **Codex plugin for Claude Code.** OpenAI shipped openai/codex-plugin-cc so Claude Code users can delegate tasks to Codex or have Codex review changes with a ChatGPT subscription [8]. Huet says the pattern they already saw in the wild was **Codex for review** and **GPT-5.4 for more complex tasks** [9]. Commands: `/codex:review`, `/codex:adversarial-review`, `/codex:rescue` [10, 11].

- **Open Codex substrate.** Huet says Codex CLI and Codex app server are open source so the same ChatGPT subscription can be used in the app, terminal, JetBrains, Xcode, OpenCode, Pi, and Claude Code [12]. The new Claude Code plugin is built on that same open-source app server + harness, including the same models, parallel tasking, and review flow [5].
- **Codex got a context upgrade.** Mark Chen says Codex now has **auto compaction**, and an early user report says it remembers tiny details across multiple rounds of compaction [7, 13].
- **Harness comparison that matters.** In Matt Maher's benchmark of frontier models implementing a 100-feature PRD, Cursor improved results from **Gemini 52→57**, **GPT-5.4 82→88**, and **Opus 77→93**; Theo highlighted Opus being **20% higher** there than in Claude Code [2, 3].
- **Local-model family to test now: Qwen3.5.** Georgi Gerganov calls it a step change across device sizes [1]. His tested local coding/chat/MCP set included **gpt-oss-120b**, **Qwen3-Coder-30B**, **GLM-4.7-Flash**, **MiniMax-M2.5**, and **Qwen3.5-35B-A3B**, mostly in **Q8_0** variants [1]. He says tool-calling quality still depends on both model intelligence and chat-template parsing in llama.cpp [1].
- **Claude Code widened enterprise support.** GitHub Enterprise Server now works across Claude Code on the web, iOS, Android, and Code Review, so self-hosted repos no longer need to move to github.com for async workflows [14, 15]. Docs: code.claude.com/docs/en/github-enterprise-server [15].
- **Claude Code added computer use, but early cost reports are rough.** Anthropic says Claude can open apps, click through UI, and test what it built from the CLI in research preview on Pro/Max plans [16]. Theo's firsthand reaction: it used up his rate limits in 2 minutes despite a $200/month plan [17].
- **LangSmith Experiments got a more useful failure view.** LangChain rebuilt the detail view to cut clutter and show better traces, clearer evaluator reasoning, and easier comparisons when debugging agent failures [6]. Try it at smith.langchain.com [6].

WORKFLOWS & TRICKS

- **Cross-model review loop from Claude Code**
  1. Install with `/plugin marketplace add openai/codex-plugin-cc` or from the repo [10, 8].
  2. Use `/codex:review` for a standard read-only pass, `/codex:adversarial-review` when you want a challenge pass, or `/codex:rescue` to hand a task off [10, 11].
  3. Keep Claude Code as your front-end if you like, but route review to Codex and heavier tasks to GPT-5.4—the pattern Huet says users were already doing manually [9].
- **Local-model bring-up: don't benchmark a broken stack**

1. Start with the highest-quality model that fits your hardware [1].
2. Use your own harness—or llama-server's webui with MCP—so you know what the stack is actually doing [1].
3. Only then optimize with quantization or community parameter tuning [1].
4. If results still look bad, inspect the whole chain: harness, chat template, prompt construction, and inference bugs [1].

- **Claude Code can build real artifacts end-to-end.** Simon Willison's flow: clone `nanochat`, pull model weights, use the Space demo source to fill in the inference script, then have Claude Code read the LLM plugin tutorial and finish the plugin [18]. The output repo is public, and Simon says it was his first full model-plugin build this way and it worked really well [18].
- **Keep secrets out of context; let the agent do the plumbing.** Kent C. Dodds says Claude Desktop cancelled a scheduled Cursor cloud agent, asked for a Cloudflare API token securely so it never entered context, generated an EC P-256 keypair, deployed a Worker, and updated Cloudflare routing to finish a Tesla integration [19]. Reusable pattern: human-mediated auth, agent-executed infra steps, MCP as the handoff surface [19].
- **If Claude Code usage suddenly spikes, test the CLI path.** Theo, relaying a reverse-engineered report he says he did **not** independently confirm, points to a standalone Bun-binary cache bug with a workaround of `npx @anthropic-ai/claude-code`, plus a separate `--resume` issue that may still break cache [20]. He says uncached tokens can be 10x-20x more expensive [20].
- **Push human effort into plan mode.** Jason Zhou says his strongest engineers now spend their time giving context and making technical decisions while the agent executes across multiple sessions [21].

PEOPLE TO WATCH

- **Georgi Gerganov** — probably the clearest current explainer of why local coding agents disappoint: harness, chat templates, prompt construction, inference bugs, and a practical bring-up order [1]. Simon Willison says this matches his own experiments [22].
- **Romain Huet** — high signal because he's shipping actual workflow glue, not just demos: the open-source Codex plugin for Claude Code, concrete commands, and the open-source Codex app server/CLI underneath [9, 10, 12].
- **Simon Willison** — published a full transcript of Claude Code building a real model plugin end-to-end; good benchmark for what a successful serious use case looks like [18].
- **Kent C. Dodds** — worth following for real MCP + infra orchestration patterns, especially his clear secret-handling boundary where tokens stay out of model context [19].

- **Jason Zhou** — useful on where coding agents meet product/design: he trained non-technical designers on Cursor + GitHub, and now ships a `/superdesign` skill that scans the codebase before designing in context [21].

WATCH & LISTEN

- **14:15-16:39 — Meta Harness's self-improvement loop.** Fastest clean explanation of the pattern: store source, scores, and traces on disk; let a coding-agent proposer inspect prior failures; iterate the harness instead of stuffing everything into one prompt [4].

AI Self EVOLUTION (Meta Harness) (14:14)

- **37:25-38:28 — Jason Zhou's `/superdesign` flow.** Nice crossover example: the agent first scans the codebase and component system, then opens the browser and designs with actual product context instead of guessing from scratch [21].

The AI Tool Built for Founders Who Can't Afford a Designer Yet w/ Jason Zhou (37:25)

PROJECTS & REPOS

- **openai/codex-plugin-cc** — open-source bridge letting Claude Code call Codex for task delegation or code review via ChatGPT subscription [8]. Signal that it solves a real behavior: Embiricos says enough developers were already using Codex to review Claude outputs that OpenAI decided to lean into it [23].
- **Meta Harness** — new open-source project from Stanford, MIT, and Krafton for end-to-end optimization of model harnesses; paper and code are already out [4]. The core design is a coding-agent proposer with filesystem access that iterates on prior harnesses instead of relying on a fixed scaffold [4].
- **simonw/llm-mrchatterbox** — a useful reference repo for the Claude Code-built-plugin pattern; Simon says it was his first full model-plugin build this way and he expects to use the method again [18].
- **Karpathy's autoresearch** — Matthew Berman cites it as a close cousin to Meta Harness, with **61k stars** and a self-improving loop that runs experiments and learns from prior results [4].

Editorial take: the fastest-moving edge in coding agents is no longer just model choice; it's the harness around the model — memory, routing, review, and *debugging visibility [1, 2, 4].*

Sources

1. X post by @ggerganov
2. X post by @edwinarbus
3. X post by @theo
4. AI Self EVOLUTION (Meta Harness)
5. X post by @romainhuet
6. X post by @LangChain
7. X post by @markchen90
8. X post by @dkundel
9. X post by @romainhuet
10. X post by @reach_vb
11. X post by @romainhuet
12. X post by @romainhuet
13. X post by @alxfazio
14. X post by @_catwu
15. X post by @katchu11
16. X post by @claudeai
17. X post by @theo
18. Mr. Chatterbox is a (weak) Victorian-era ethically trained model you can run on your own computer
19. X post by @kentcdodds
20. X post by @altryne
21. The AI Tool Built for Founders Who Can't Afford a Designer Yet w/ Jason Zhou
22. X post by @simonw
23. X post by @embirico