Theo's "Council" skill: models from different families must score a change before the agent stops

Coding Agents Alpha Tracker

2026-11-16

Theo's "Council" skill: models from different families must score a change before the agent stops

By Coding Agents Alpha Tracker • November 16, 2026

Theo's Council skill brings models from different families in to review changes and won't stop until they give a set score. Also: Steve Yegge's case for slowing down, Kent C. Dodds' software-factory tips and remote agent setup, and a larger context window for Gemini models in Antigravity.

Theo's Council: reviewers from different model families, with a score to hit

In "Stop Searching For the Perfect Model," Theo argues against routers that pick the right model for each task. He mostly stays on Opus 5.5 and switches only when he has unused tokens on another subscription [1]. Where he does use several models is review. For hairy changes, his **Council** skill has models from different families review the work. His reasoning is that Anthropic's 5.5 models are distilled from the same base, so asking Opus and its siblings to review one another is "like asking the same person at different times of the day" [1].

How he runs it: - He gives a prompt like `fix X, Y, and Z. Full stop with council 9.5`. "Full stop" tells the agent to do everything it can to reach a PR it is confident can be merged, then stop before merging. "Council 9.5" means the work isn't done until every reviewer scores it at least 9.5/10 [1]. - T3 Code doesn't depend on any one harness, so one harness can call another. Opus starts reviewers: Claude models in Claude Code, OpenAI models in Codex, and Muse in its own harness. Scores go at the bottom of the PR [1]. - The reviewers aren't just reading a diff. They can run and test the code on the same machine and

ask the parent agent questions [1]. - You can change what the score measures. He has asked for scores on “how likely is this to piss Theo off,” and on whether a PR is what a specific user, Corey Quinn, would want. He also tells agents they may scrap the PR, because “agents are very scared” to delete code [1].

His results: across 144 reviewer scores, Opus’s own rating of its changes rose from an average of 7.4 to 8.9 after two or more rounds. He notes that even reviewing with Opus alone would probably improve the code [1]. Bugs it caught include three security holes in T3 Code’s guest-share feature, such as the browser cache still serving visuals after a share was revoked [1]. It also flagged test deletions that removed guards against old bugs [1]. He estimates about an 80% chance he can merge a PR his threads open [1].

Caveats: in a few runs the council never agreed. One went 50+ loops, started 600+ subagents, and grew a PR of about 1,000 lines to more than 20,000. He has since added a run cap and an instruction not to let repeated reviews cause scope creep. He says Opus respects it, but OpenAI models keep going after every small nit [1]. He also lowered reviewer effort from extra-high to high [1]. He uses up Muse’s weekly limit on the $200 plan in two to three days, and says not to use the skill if you pay API prices [1]. The skills are now in a GitHub repo linked in the video description [1].



Stop Searching For the Perfect Model (8:31)

His sponsor segment has a related idea. Macroscope lets T3 Code keep a folder of custom review agents in the repo, scoped by file type. One holds the maintainer’s

list of things that annoy him in agents' Effect code. Another stops agents from creating UI components that already exist [1].

Yegge: pace agents to human speed, and think twice before a factory

Steve Yegge summed up his conference talks in a post. His agents have overwhelmed his game's players with features, his accountant with emails, his merge queue with tickets and his machines with load. They have also "launched broken OSS without permission" [2]. He says you probably don't need a software factory: he has had to throw away five because rewriting them became cheaper than fixing them, and keeping one running takes real money and leadership [2]. His advice for now is to build your own agentic chief of staff and get good at working with it [2].

Kent C. Dodds offers two prompts for those who do build a factory: - Ask agents to complain about what they hate about working in your codebase [3]. - Before sending any prompt, ask "how could I have made it so I don't have to send this myself and still get the same desired outcome?" [4]

Running agents on a home machine, and the credentials problem

From a New York hotel, Dodds used Kody to install T3 Code on his home NAS. That lets him start Claude Code agents while his laptop is off, without Tailscale [5, 6]. He reported it "worked," though a screenshot shows Docker Hub still failing with a certificate error [7]. Self-hosting Kody is free for personal, nonprofit, noncommercial and classroom use, and comes with a 30-day evaluation, on Cloudflare or via Docker [8].

A related open question: Marc Köhlbrugge asked how to give coding agents credentials securely, noting that 1Password gives an agent access to the full vault every time [9]. Dodds answered that this is the problem he built Kody for, but didn't explain how Kody handles it [10].

Model and tool updates

- **Antigravity:** the context window doubled for Gemini Argon, Carbon and Hydrogen. The team says compaction should drop 3.5x per user turn [11]. Nicholas Moy, from the Antigravity team, says Carbon is still "only ok" at swarms and wants reports on where it falls short [12].
- **Codex:** Tibo says the team "might have actually fixed" the "model at capacity" errors [13].
- **Fable 5.2 vs. Opus 5.5:** ThePrimeagen says his work is too trivial to tell Fable 5.2 from Opus 5.5 high [14]. That fits last period's advice to keep Opus as the default.

- **Agent UI:** Theo says kanban-style agent dashboards waste over half the screen on in-progress work [15]. He adds that Vibe Kanban, which he invested in, was moving toward a T3 Code-style layout before it shut down [16].

Sources

1. Stop Searching For the Perfect Model
2. X post by @Steve_Yegge
3. X post by @kentcdodds
4. X post by @kentcdodds
5. X post by @kentcdodds
6. X post by @kentcdodds
7. X post by @kentcdodds
8. X post by @kentcdodds
9. X post by @marckohlbrugge
10. X post by @kentcdodds
11. X post by @_mohansolo
12. X post by @thenickmoy
13. X post by @thsottiaux
14. X post by @ThePrimeagen
15. X post by @theo
16. X post by @theo