Opus 5.5 and GPT-6 Sol/Luna Put Coding-Agent Economics to the Test

Coding Agents Alpha Tracker

2026-09-23

Opus 5.5 and GPT-6 Sol/Luna Put Coding-Agent Economics to the Test

By Coding Agents Alpha Tracker • September 23, 2026

Anthropic's Opus 5.5 and OpenAI's GPT-6 Sol/Luna arrived with lower-cost claims and early coding results; a first-party HAProxy migration gives the clearest concrete signal. The brief pairs model economics with practical workflows for effort selection, PR completion, formal verification, and agent tool search.

TOP SIGNAL

Compare cost per completed change, not token rates alone. Anthropic says Opus 5.5 matches Claude Fable 5.1 for most tasks at 40% lower cost on typical workloads; it is now the medium-effort default in Claude Code and the Claude app for Pro, Max, and Team. OpenAI's GPT-6 Luna starts at $0.10 per million input tokens and $0.50 per million output tokens. [1, 2, 3, 4]

The strongest coding datapoint is Boris Cherny's internal HAProxy C-to-Rust test: Opus 5.5 and Fable 5.1 both passed nearly all tests, but Opus 5.5 finished in 9.5 hours versus 12 and cost 51% less. Treat it as a promising first-party result, not an independent bake-off. [5, 2]

TRY THIS

- **Sweep effort settings on a real task.** Matthew Berman's read of Anthropic's FrontierCode comparison says Opus 5.5 at medium effort scored higher than max while costing under $1 versus over $5 per task. Run a representative repo task at both settings and compare test results, elapsed time, and cost before pinning a default. [6]
- **Give long-running coding work an explicit end state.** In Theo's T3 Code workflow, the prompt supplied a screenshot and thread ID, asked the

agent to find root causes, fix them, and file one focused PR. Say whether it should stop at the PR, babysit it until CI is green, or merge once green; Theo's babysit instructions check only comments and CI newer than the latest push, verify bot findings, and avoid scope creep. [7]

- **Try an agent-assisted formal-verification pass on risky code.** Boris Cherny says a couple of short prompts with Opus 5.5 and Lean produced 16 PRs fixing bugs and race conditions in the Claude Agent SDK; he sometimes combines Lean and TLA+ to probe data flow, concurrency, and state management. [8]
- **For tool discovery, keep lexical recall broad and rerank candidates with Jev.** Treg takes the top 30 lexical matches on any rare query term, asks Jev whether each tool would directly accomplish—or be necessary for—the task, drops scores below 0.4, and prioritizes scores of 0.7 or higher while retaining lexical order and existing success/price adjustments. Its interleaving favored the new pipeline by 8.6:1 points excluding ties—not 9× search accuracy—and the test combines broader retrieval with Jev, so it does not isolate Jev's contribution. [9] Treg's write-up.

WHAT SHIPPED

- **Claude Opus 5.5:** Anthropic lists API rates of $4/$20 per million input/output tokens and $0.20 per million cache reads—20% lower input/output prices and 60% cheaper cache reads than Opus 5. It claims 40% lower cost on typical workloads at default settings and over 30% faster output; Anthropic also cautions that benchmark margins are a less reliable guide to real-world differences than before. [2]
- **GPT-6 Sol and Luna:** OpenAI lists Sol at $2/$10 and Luna at $0.10/$0.50 per million input/output tokens. Its DeepSWE 1.1 results report Sol at 68.8% on max effort, within 1.1 points of Claude Fable 5 at xhigh and at about 80% lower task cost; Luna scores 66.6% at max, described as comparable to Opus 5 and Fable 5 at medium effort. These are OpenAI-reported evaluations; validate them in your own harness. [4, 10]
- **LangChain's Patch demo** turns a Slack feature request into a GitHub PR ready for review. The Managed Deep Agents recipe uses `agent.py` for the agent/model, `instructions.md` for the repo and procedure, a Slack channel, GitHub MCP with its token in an environment-variable secret, and `define_sandbox` for coding and tests. [11, 12]
- **Simon Willison's LLM CLI updates:** `llm 0.36` adds `gpt-6-sol` and `gpt-6-luna`, plus a guard for single-turn models; `llm-anthropic 0.29` adds Opus 5.5 (`llm -m claude-opus-5.5 "prompt goes here"`); `llm-typesafe 0.1a0` adds Jev support (`llm install llm-typesafe`). [13, 14, 15]

GO DEEPER

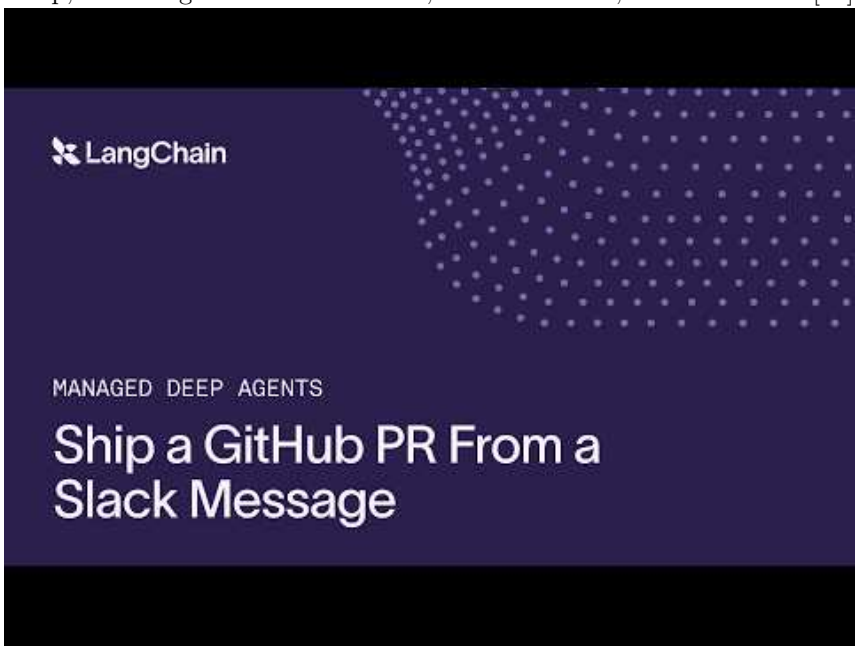
- **Matthew Berman on Opus 5.5's effort economics:** The useful segment compares effort settings and cost per task; watch it before assuming max is the best default. [6]

Anthropic went CRAZY (Opus 5.5) (6:04)

- **Theo on long-running Fable coding work:** A firsthand PR workflow—context, root cause, review-bot feedback, and a defined stop condition. [7]

I was using Fable wrong, this is how I fixed it (14:36)

- **LangChain's Patch walkthrough:** See the Slack-to-PR agent's setup, including its instructions file, GitHub access, and sandbox. [12]



Ship a GitHub PR From a Slack Message with Managed Deep Agents (1:29)

- **Repo — `llm-typesafe`:** A quick way to inspect Jev's typed yes/no, choice, and scoring calls from the LLM CLI; the release includes install and API-key setup. [15]

Editorial take: Cheaper models widen the options; effort tuning, explicit PR completion criteria, and verification determine whether the savings turn into shippable code. [6, 7, 8]

Sources

1. X post by @claudeai
2. Introducing Claude Opus 5.5
3. X post by @_catwu
4. Claude Opus 5.5, GPT-6 Sol, GPT-6 Luna, and a new price war
5. X post by @bcherny
6. Anthropic went CRAZY (Opus 5.5)
7. I was using Fable wrong, this is how I fixed it
8. X post by @bcherny
9. X article by @SToneoneX
10. Introducing GPT-6 Sol and Luna | OpenAI
11. X post by @LangChain
12. Ship a GitHub PR From a Slack Message with Managed Deep Agents
13. llm 0.36
14. llm-anthropic 0.29
15. llm-typesafe 0.1a0