

THE SHIFT UNDERWAY IN 2026

Vibe Coding Is Over. Orchestrating Agents Is **The Job Now.**

Karpathy renamed his own term within a year. The productivity data explains exactly why.

Four things worth ten minutes

- 01** The renaming: what Karpathy called it in 2025, and what he calls it now
- 02** What happened in eighteen months, at Google and at Anthropic
- 03** Why the productivity data is split, and what it's actually measuring
- 04** Four habits to build before this becomes the baseline expectation

Feb 2025 → 2026

Andrej Karpathy coined "vibe coding" in February 2025. Within a year he renamed his own practice "agentic engineering."

His own words: "the new default is that you are not writing the code directly 99% of the time, you are orchestrating agents who do and acting as oversight." That's not a rebrand. It's an admission the first version was never the destination.

75%

of new code at Google is now AI-generated, up from 50% just six months earlier

Sundar Pichai, Google Cloud Next 2026. Every line is still reviewed and approved by engineers. A complex migration ran six times faster than a year earlier; a macOS app went from idea to working prototype in days, both built through orchestrated agents, not solo prompting.

-19% → -4%

Experienced developers using unstructured AI assistance got slower, not faster, in METR's 2025 randomized trial

A February 2026 follow-up with a larger, more diverse cohort narrowed the gap to roughly breakeven. Developers in the original study believed they were 20% faster. They were 19% slower. Perception and reality moved in opposite directions.

60 Claude subagents pushed a 37-year-stuck math bound from 41.6% to 67.2%

650

DEAD-END IDEAS DISCARDED

2

AGENTS FOUND THE WINNING APPROACH

13

CONTRIBUTED PIECES TO IT

13

DID NOTHING BUT TRY TO DISPROVE IT

Verified by Anthropic mathematicians, external reviewers, and a machine-checked Lean proof before anyone called it done. That's the director's job, at the extreme end.

What actually changes

DIMENSION	VIBE CODING	DIRECTOR MODE
Unit of work	A prompt	A spec or a graph of roles
What gets reviewed	The diff, if anyone looks	The plan, the trace, the verification gate
Failure mode	Wrong code accepted silently	Caught by a dedicated validator role first
Parallelism	One conversation, one thread	Dozens of subagents, distinct roles



The review unit is moving from the diff to the spec and the verification gate.

That's the whole shift, in one sentence.

Four habits to build this quarter

- 1 Write specs, not just prompts, before any code gets generated
- 2 Review the trace and the plan, not just the diff
- 3 Build a verification gate before you expand what an agent can touch
- 4 Practice decomposition: planner, implementer, checker, as a deliberate skill

The debate is over. Not
because anyone won it.
Because the person who
named it **moved on.**

Read the full piece, with sources, on terencekok.com.



Terence Kok

AI & Innovation Strategist

SAVE · COMMENT · FOLLOW