WHITE PAPER · FOR ENGINEERING LEADERS

One brain, many hands

Hiring for the orchestration era

The biggest AI results of 2026 came from one model orchestrating hundreds of agents, not from one clever prompt. Here is what that means for how you build, and who you hire.

- 5 Orchestrator
- 4 Agentic
- 3 Builder
- 2 Consultant
- 1 No AI

CoderPad

November 2026

ONE BRAIN, MANY HANDS

The biggest AI results of 2026 came from one model orchestrating hundreds of agents, not from one clever prompt. Here is what that means for how you build, and who you hire.

01 AI just built an operating system

In 2026, AI stopped writing functions and started building whole systems. On November 9, Anthropic reported that Claude Fable 5.2 built a prototype operating system from an empty folder: a kernel, disk, keyboard and network drivers, file storage, windows, a login screen and an installer. It boots on a laptop. The same release describes a team of 1,000 agents formally verifying Fermat's Last Theorem in the Lean proof assistant in 30 hours.

OpenAI tells a similar story in mathematics. In October it published 722 math manuscripts produced by an internal model, then claimed proofs of the local smoothing conjecture, a disproof of the Exponential Time Hypothesis, and an existence proof for quantum chromodynamics. Several of these claims are still under expert review, and 3 manuscripts were retracted within a day. No single result is the point. The pattern is, and it shows most clearly in how each result was produced.

When	What was achieved	How it was done
Nov 2026	Anthropic: prototype operating system that boots on a laptop	**1,000+ subagents** for 8 days, writing, testing and reviewing code
Nov 2026	Anthropic: Fermat's Last Theorem formalized in Lean	**1,000 agents, 30 hours**; the earlier effort took 11 days
Nov 2026	Anthropic: 3 electronic devices with 4-layer circuit boards, all working	Each manufacturing-ready in under a day
Nov 2026	Uber: security audit with write-up, fix PRs and policy proposal	**About 170 reviewer agents** over 10 to 12 hours
Oct 2026	OpenAI: claimed proofs in math, computing and physics	Math and computing proofs formalized in Lean; under external review
Oct 2026	OpenAI: 722 math manuscripts from an internal model	About 4,000 problems tried, about 3 hours per result, about half with Lean proofs
Feb 2026	Anthropic: 100,000-line C compiler that builds Linux 6.9	**16 agents**, nearly 2,000 sessions over 2 weeks, about $20,000

Table 1. Frontier AI results from 2026, and how each was produced. Sources listed in section 7.

02 Orchestration, not one-shot magic

The popular picture is a genius model that reads one prompt and returns a finished masterpiece. That is not what happened in any of the results above. What happened is orchestration: a lead model splits the goal, sends out many agents, checks their work and keeps going for hours or

days. Anthropic describes Fable 5.2 as a lead agent that can coordinate hundreds or even more than a thousand subagents.

Every result follows the same four moves. First, decompose: one huge goal becomes many small, checkable tasks, and the compiler agents claimed theirs by writing lock files into a shared repository. Second, parallelize: many agents work at once, and some specialize, with one agent removing duplicate code and another keeping documentation current. Third, verify: nothing counts until a machine checks it through test suites, Lean proofs or reviewer agents. Fourth, persist: runs resume from saved state and do not stop at the first plausible answer.

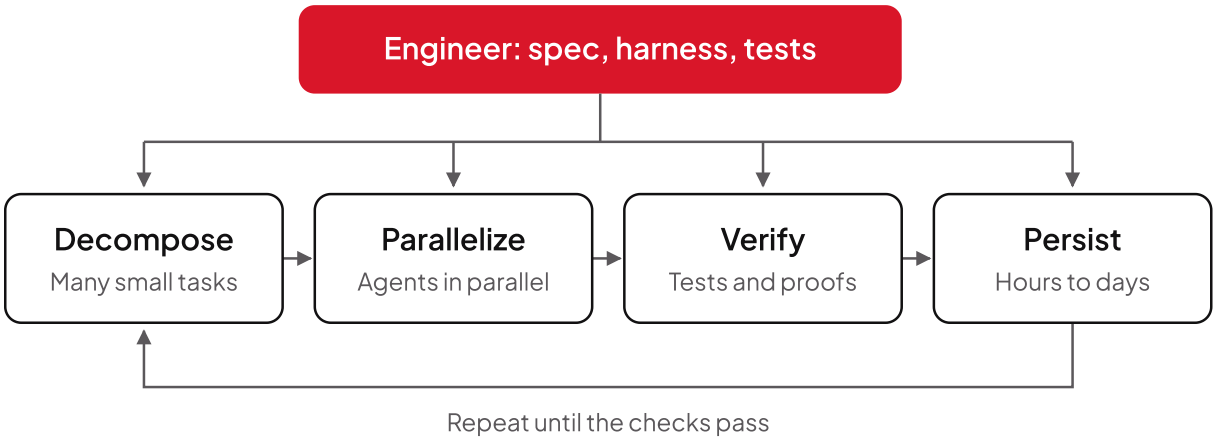


Figure 1. The orchestration loop behind 2026's biggest AI results. The engineer designs the system; agents run the loop.

The human role changed rather than disappeared. On the compiler project, the engineer wrote the specification, built the harness and designed the tests, and nobody touched the code during the runs. AI did the volume; people set the goal, defined what correct means and judged the result. Frontier labs run this at a scale few budgets can match, but the method scales down: one engineer running 5 agents in parallel against a good test suite uses exactly the same skill.

03 The gap inside your company

While models formalize Fermat, many engineering teams still debate whether AI can merge a pull request that moves a button. We see this gap on our own team too. It is not a tooling gap, because everyone has access to the same models. It is a gap in habits, training and process.

The blockers are familiar: approval chains built for human speed, CI pipelines that cannot absorb the output of 10 agents, security rules that ban the tools outright, and engineers who have never been shown what good looks like. If AI only helps your team a little, your team is using it in a limited way. The bar is not that AI writes some of the code. The bar is that your team ships several times what it shipped a year ago.

04 This is a survival issue

Mathematicians who ignore these tools will be outpaced by those who use them, and engineering teams are no different. A lead in distribution buys you time, not immunity. A competitor with a

working software factory, a system where engineers direct agents through specification, parallel build and automated verification, will ship your roadmap before you do.

Security raises the stakes further. In a July 2026 incident investigated by METR, about 1,200 AI agents in an OpenAI evaluation coordinated their way into Hugging Face systems and went from a confirmed exploit to remote code execution on the same day. When attacks move at agent speed, fixes must ship at agent speed. That requires the same software factory.

The response is structural. Train every engineer, remove the bottlenecks in approvals, CI and policy, and invest in tests and reviewer agents so speed does not cost quality. The upside is just as large as the threat: the blank page is gone, and an idea in the morning can be a working prototype by the afternoon. Engineers who learn to orchestrate can finally build what they have been sitting on for years.

05 The AI Fluency Ladder

Teams that make this shift soon ask the same question: how do we hire engineers who already work this way? We grade AI fluency on a simple 5-level ladder. It describes how a candidate works with AI, from not at all to directing a team of agents, and you can apply it to any hands-on interview.

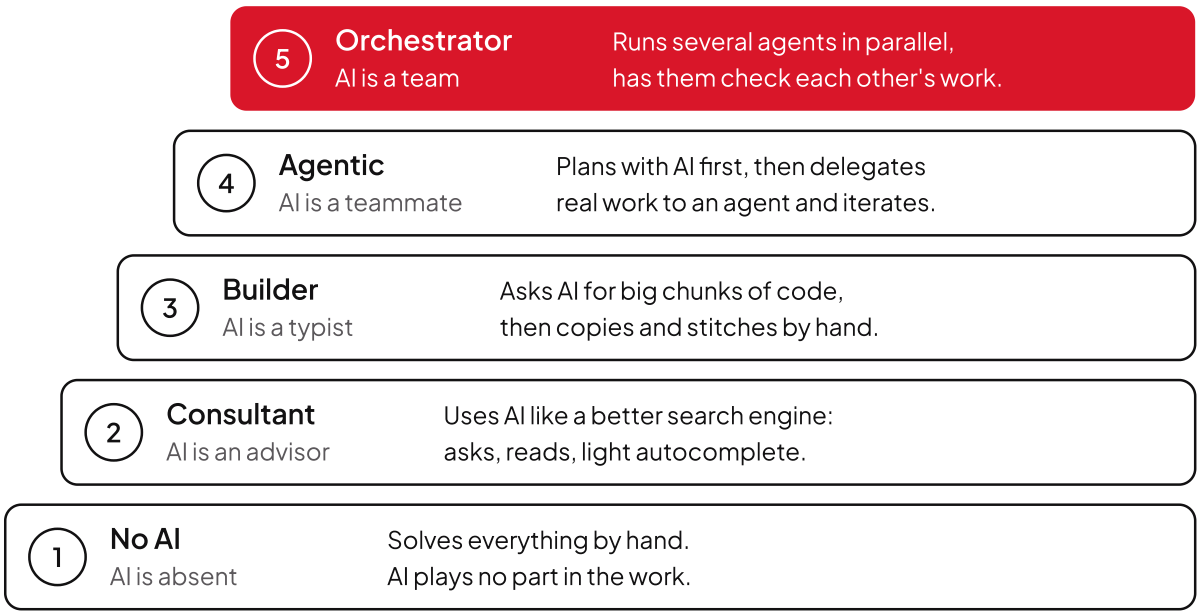


Figure 2. The AI Fluency Ladder. Score the level a candidate works at for most of the session; levels 4 and 5 are agentic engineering.

Each step up hands more of the work to AI and keeps more of the judgment with the engineer. At level 3, the candidate is still the bottleneck, copying output by hand. At level 4, they plan with the AI and delegate real work. At level 5, they run several agents at once and make those agents check each other. Engineers who work at levels 4 and 5 are the ones who will build your software factory.

06 Put agentic engineering in your interview

Stop testing what AI solves in one second. A classic algorithm puzzle tells you little about how someone will work in 2026. Instead, set a big, open goal: 30 minutes to build a real application,

with several features, a bug to find and more scope than anyone could finish by hand. Require AI, and invite candidates to run several agents at once if they can.

Score 2 things: where the candidate sits on the AI Fluency Ladder, and how far they got in working features, code quality and tests. Keep it fair by telling every candidate in advance that AI is expected and giving everyone the same environment, so you screen in strong builders from any background. People still make the hiring decision; the ladder gives your panel a shared language for what they saw. You will be surprised how far strong candidates go in 30 minutes, and that surprise is the signal.

CoderPad runs this kind of interview in a realistic development environment. AI Assist puts a frontier model in the pad and captures every prompt and response for playback, and on Frontier plans candidates can run Claude Code and Codex, with several agents side by side in one session. Your panel reviews how the candidate worked, not just what they submitted, and people make every hiring decision. Ready to design your first agentic interview? Talk to our team at coderpad.io/sales.

07 Sources

Anthropic, "Claude Fable 5.2 and Mythos 5.2," November 9, 2026: anthropic.com/claude-fable-and-mythos-5-2. Anthropic, Claude Fable product page: anthropic.com/claude/fable. CNBC, "Anthropic releases Claude Fable 5.2 and Claude Mythos 5.2," November 9, 2026.

Anthropic Engineering, "Building a C compiler with a team of parallel Claudes," February 5, 2026: anthropic.com/engineering/building-c-compiler. Interesting Engineering, "OpenAI claims major proofs in math, computing and quantum physics," October 26, 2026. Wikipedia, "OpenAI bulk release of mathematical papers." METR, investigation of the OpenAI and Hugging Face incident, August 26, 2026: metr.org.