

Developers want agents that build software without them. Four problems stand in the way.

User intent is fractal.

Every detail you specify hides finer ones.

User intent is latent.

You cannot fully say what you want without building it.

User intent is dynamic.

Seeing the work changes what you want.

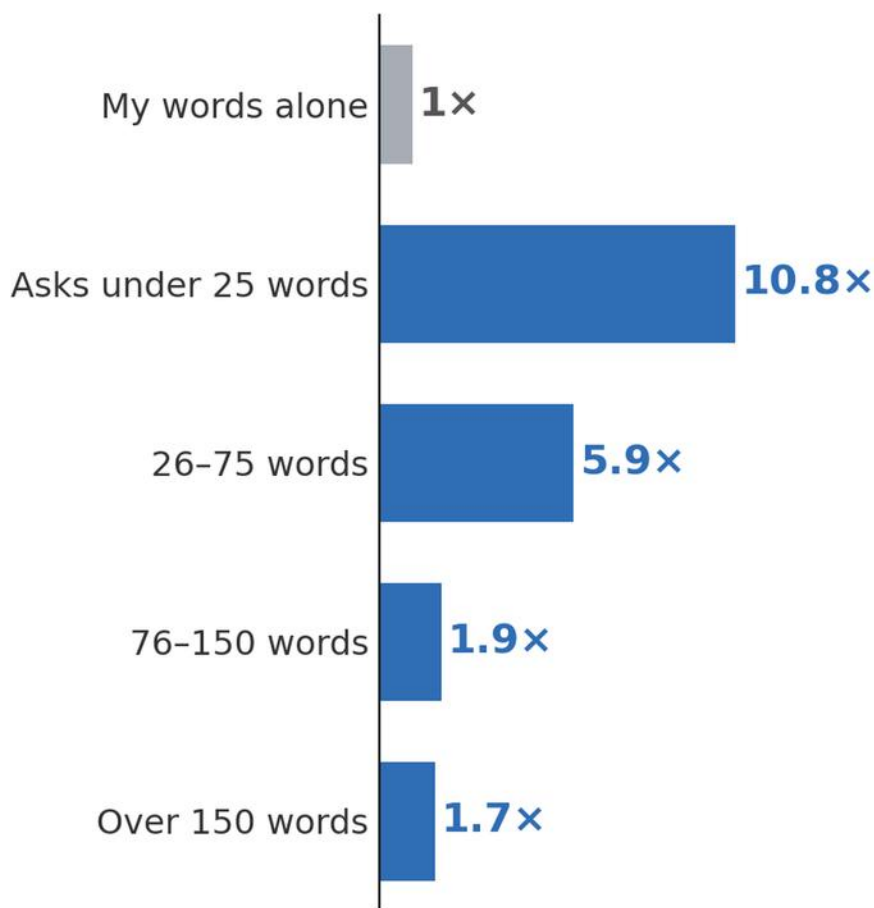
Agent performance is unverifiable.

The more you delegate, the less you can check.

Agents can shrink the human's part, never remove it.

A case study: 39 days running my own home agent system (about 2,400 messages, 1,794 merged pull requests).

For short requests, the agents wrote 11× more words of “what I meant” than I did.



866 requests, Days 6-26: my verbatim words 45,596; words written for me 217,734 (4.8× overall). Median per request.

The machine's check said every worksheet was recorded. The tablet had recorded none of the 28.

Without the head: the machine's check (emulator)

3 of 3 recorded

With the head: read on the real tablet

0 of 28 recorded

Days 25–26: the emulator check (3 of 3) against the real tablet's own record (0 of 28). Only reading the device caught it.

WHY FULLY AGENTIC DEVELOPMENT IS UNSOLVABLE

Two models writing down what “done” means for the same requests agree on only about half of it.

Astra, written twice

● 72.8%

Gemini, written twice

● 67.2%

Astra vs Gemini

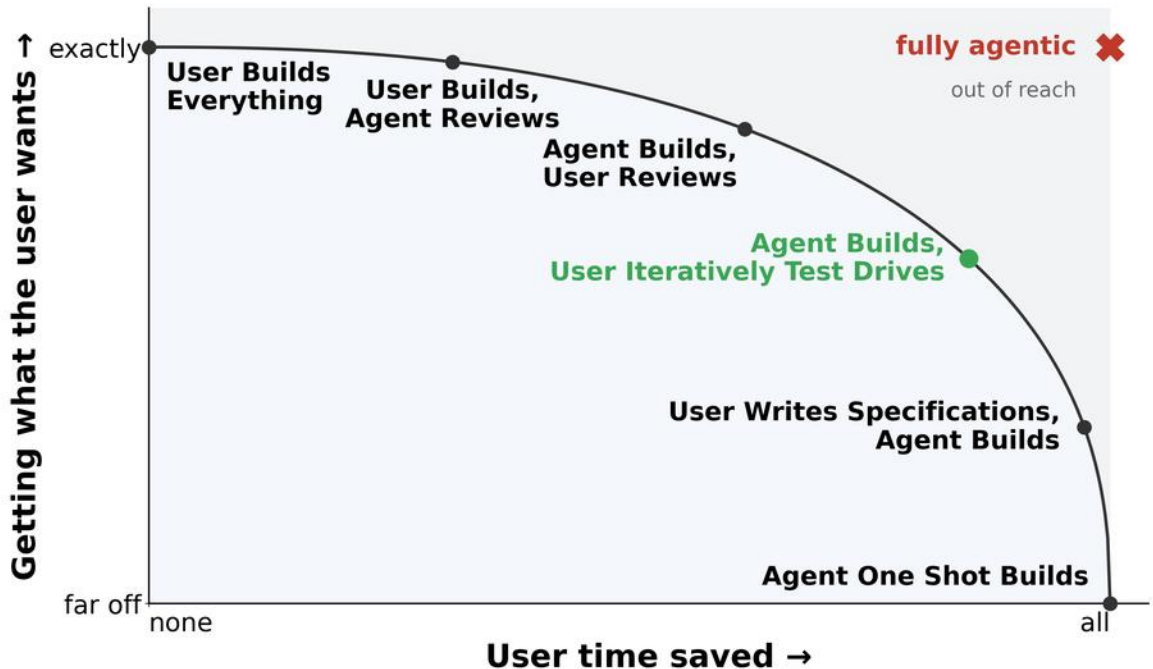
● 53.3%



434 requests, Day 24: the same model twice 67-73%, two different models 53%. Overlap judged by a third model; 95% intervals.

WHY FULLY AGENTIC DEVELOPMENT IS UNSOLVABLE

There is no fully agentic point: only a frontier between the user's time and getting what they want.



Conceptual. Points on the curve are ways of working; ours (green) is the agent builds while the user iteratively test drives.

An implied research agenda

1 Efficiently and correctly soliciting intent

What is the fewest questions that pins down what the user wants?

2 Measuring the fractal and dynamic level of intents

Can we tell, before building, how deep an intent goes and how fast it will move?

3 Measuring the cost of verifiability

What does checking cost, and who pays it: a machine, the user, or a hired expert?

Full talk and slides at rexdouglass.com.