

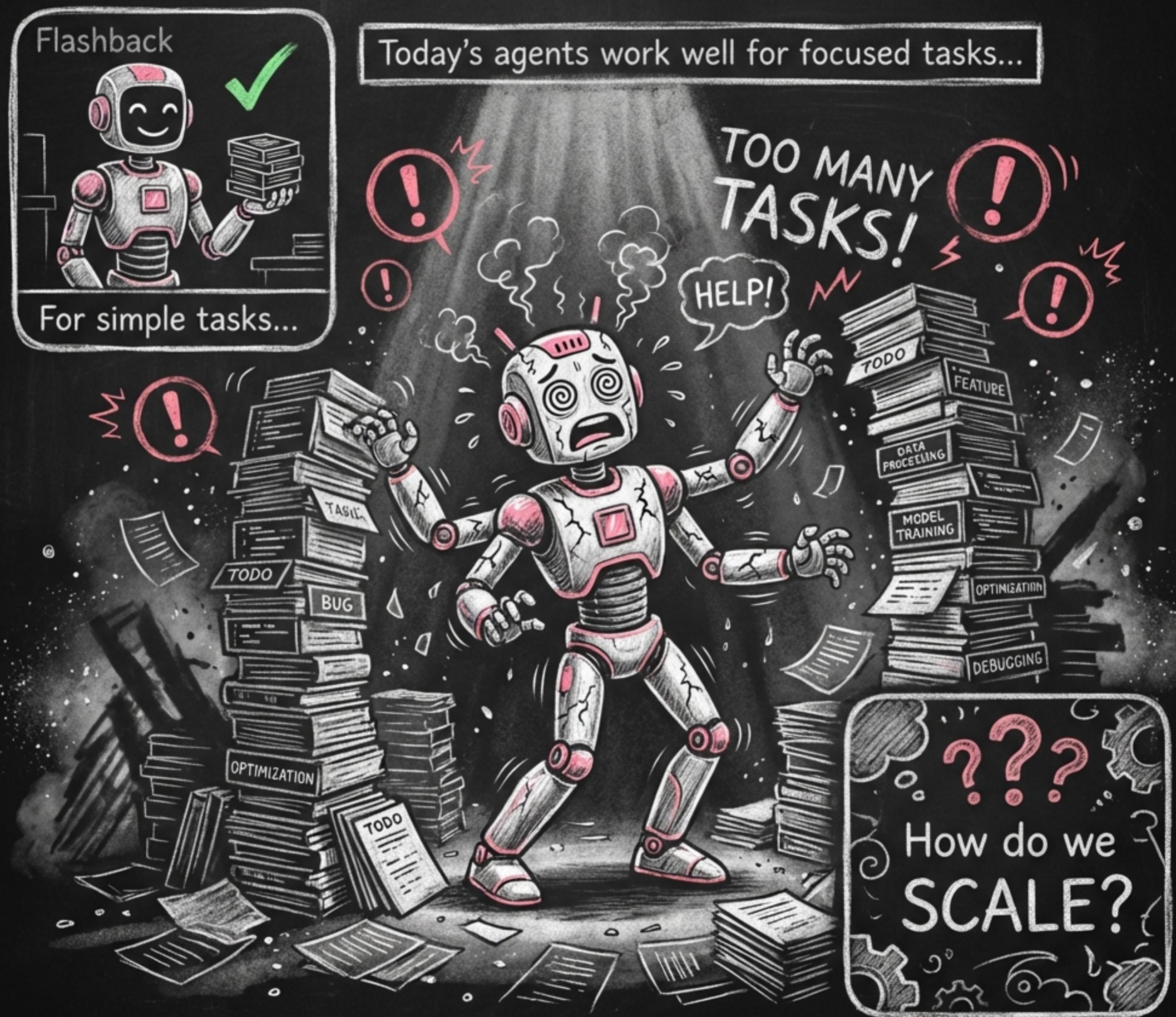
THE AGENT ORCHESTRA

Scaling Autonomous Coding



Based on Cursor research

Chapter 1: The Limits of a Single Agent



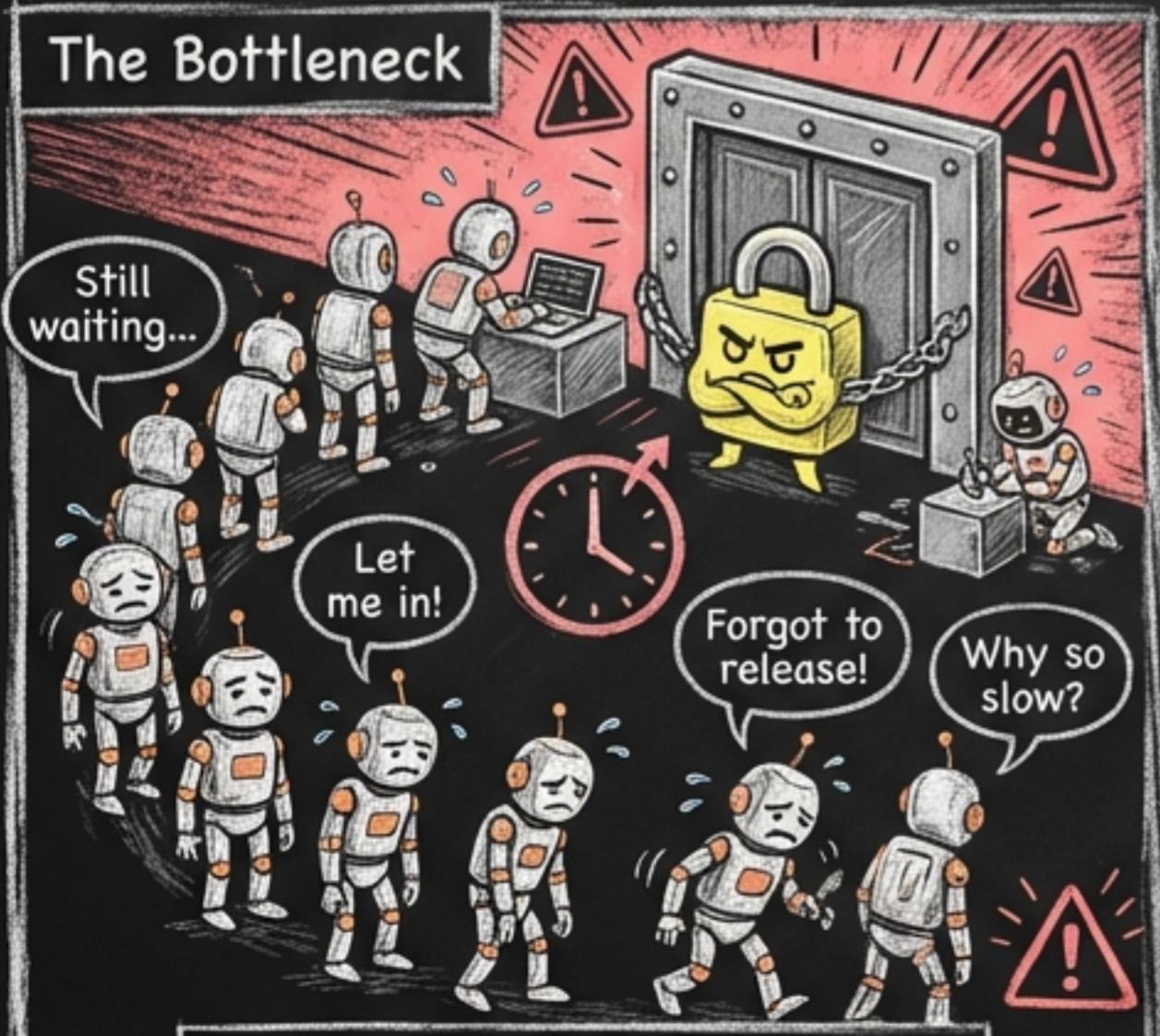
First attempt: Let agents self-coordinate through a shared file



The Lock Appears



The Bottleneck



Twenty agents slowed to the throughput of two or three

The Crash

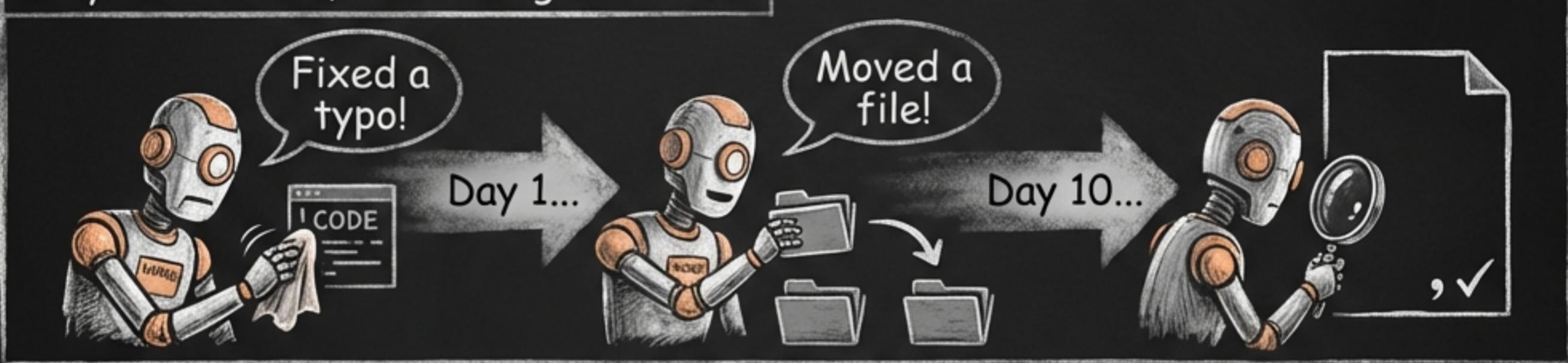


Agents would hold locks too long, forget to release, or fail while holding them

With no hierarchy, agents became risk-averse



Only Small Safe, safe changes instead



Hard Problems Remain



THE BREAKTHROUGH

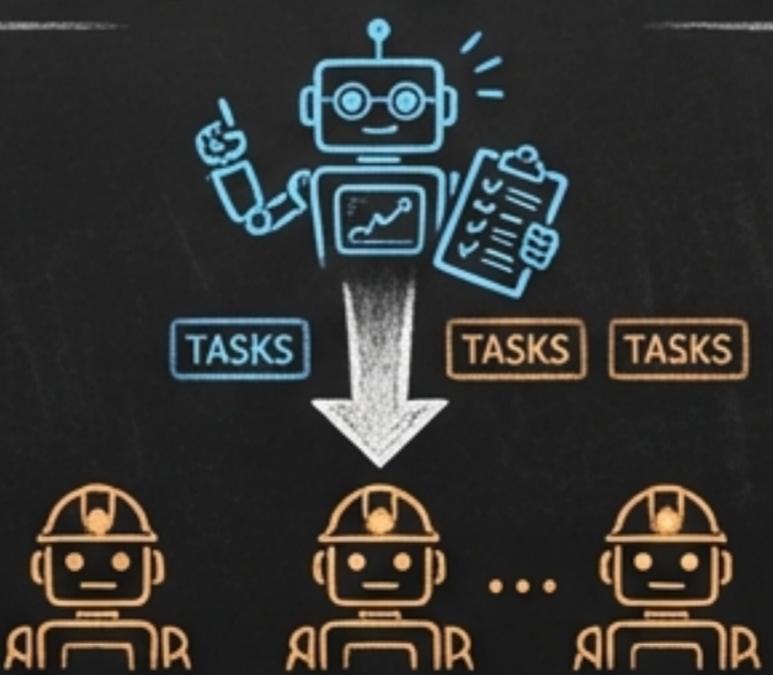


PLANNER

Explores codebase, creates tasks, spawns sub-planners

WORKERS

Pick up tasks, focus on completion, push changes



- ✓ No coordination overhead
- ✓ Clear ownership
- ✓ Parallel scaling

DAY 1

Building a web browser from scratch



- HTML Parser
- CSS Engine
- JavaScript Runtime



PLANNER

WORKER

CODE MOUNTAIN

DAY 3

CLICK!

100+
agents

1000+
files

PUSH!

MERGE!
COMPILE!

CODE MOUNTAIN

WEEK 1 COMPLETE

1 MILLION
LINES OF CODE

CODE MOUNTAIN

It Works!

BROWSER WINDOW

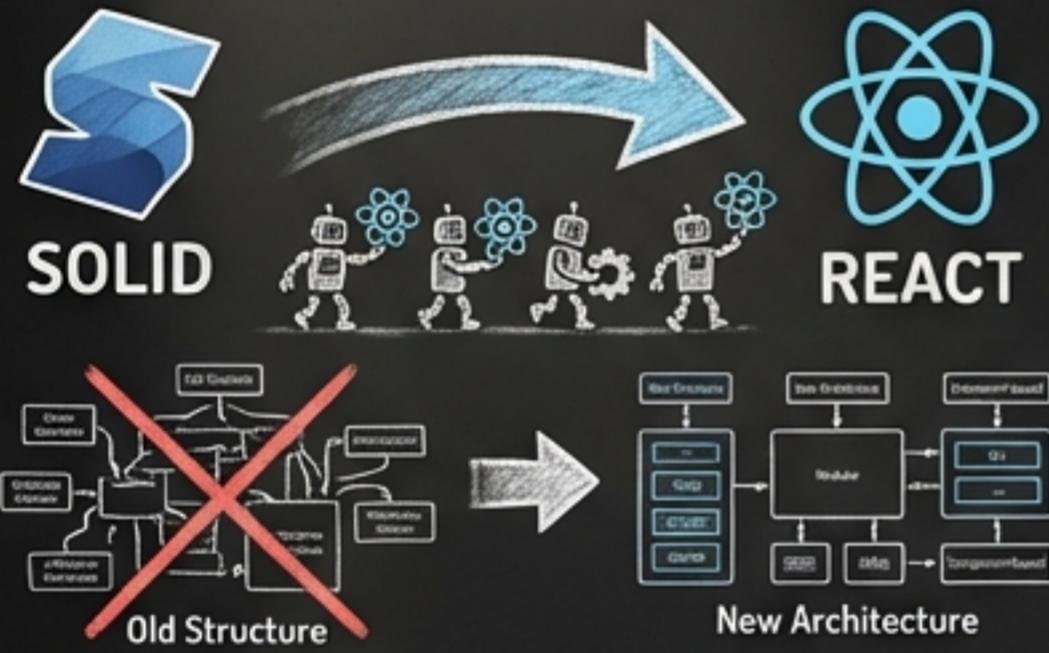


Despite the codebase size, new agents can still understand it and make meaningful progress

TO BE CONTINUED...

TRILLIONS of tokens WEEKS of runtime

SOLID → REACT Migration



+266K / -193K edits 3+ weeks
Cursor codebase Passing CI

VIDEO RENDERING Speedup



JAVA LSP



7.4K commits
550K LoC

WINDOWS 7 EMU



14.6K commits
1.2M LoC

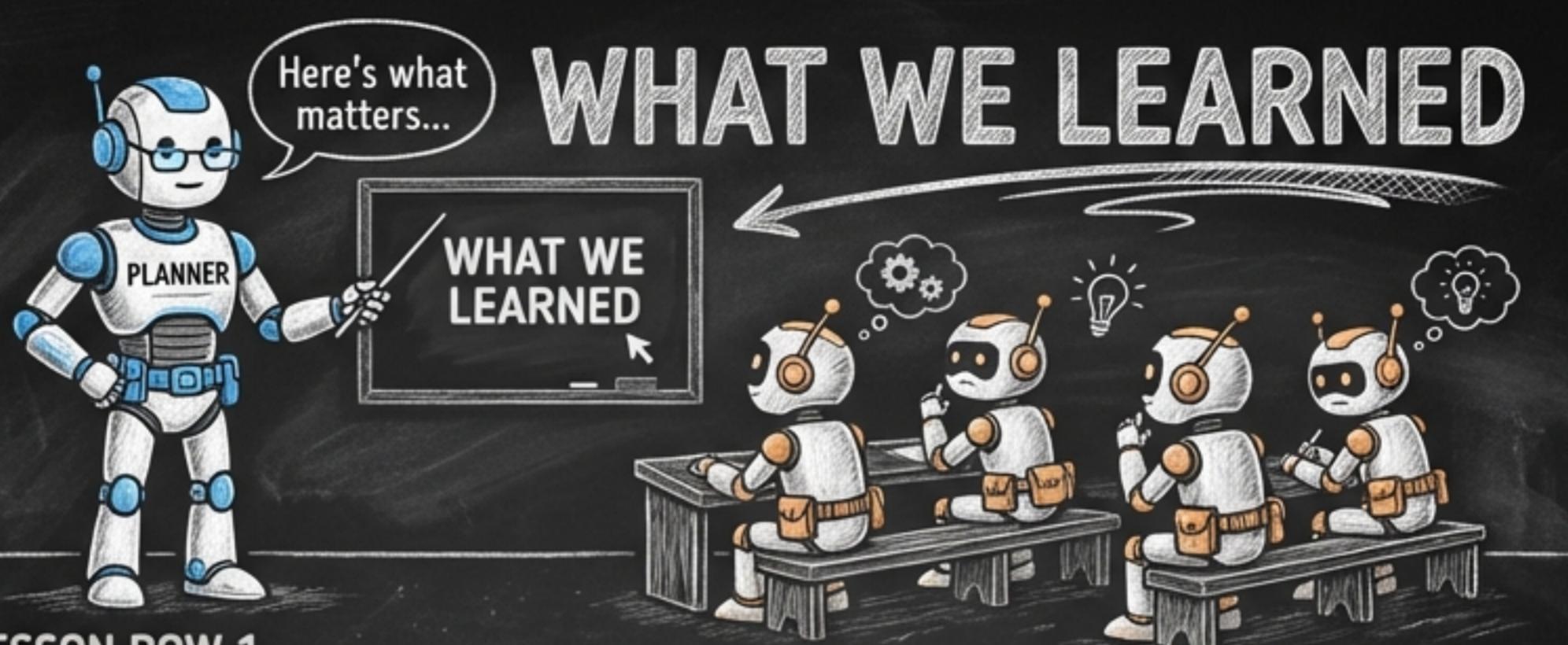
EXCEL CLONE

	6	9	6	0	1
1					
2	0	0	0	0	0
3	00	00	00	00	00
4	000	000	000	000	000
5					
6					
7					
8					
9					
10					

12K commits
1.6M LoC



Still running... →



LESSON ROW 1

1. MODEL CHOICE MATTERS
GPT-5.2 better at extended autonomous work

2. DIFFERENT MODELS FOR DIFFERENT ROLES
Best planner ≠ Best coder

LESSON ROW 2

3. REMOVING COMPLEXITY HELPED
Integrator role removed – workers handle conflicts themselves

4. RIGHT AMOUNT OF STRUCTURE
Too little = chaos | Too much = fragility

LESSON 5 - THE BIG ONE

5.

PROMPTS MATTER MOST

“A surprising amount of the system's behavior comes down to how we prompt the agents”

Hundreds of agents can work together on a single codebase for weeks

Real progress on ambitious projects



Multi-agent coordination remains a hard problem. We're nowhere near optimal.

The Question Has a More Optimistic
Answer Than We Expected

Can we scale autonomous coding by
throwing more agents at a problem? Yes.

