

AI makes teams ship faster, but decisions and outcomes haven't caught up

PM Daily Digest

2026-10-03

AI makes teams ship faster, but decisions and outcomes haven't caught up

By PM Daily Digest • October 3, 2026

Marty Cagan and Atlassian survey data both point to a gap between AI-driven output and real outcomes. OpenAI's Dots raises design questions for agent-facing products, and SaaSTr warns that charging agents for API access can backfire.

Faster output, same outcomes

At ProductTank Berlin, Marty Cagan pointed to what McKinsey calls the “AI productivity paradox.” Teams using AI tools move faster, “but no, they are not getting any real different results... the output gets a lot faster, but the outcome is still hard” [1]. His main warning is about companies still using the “project model,” where work is driven by feature roadmaps with dates. In that model, a PM's or designer's tasks can be handed to agents, and he expects those companies to end up with only engineers [1]. In the “product model,” faster delivery shifts the focus to strategy and to the craft of finding solutions that win in crowded markets: “the cheaper delivery becomes, the more important the craft and strategy becomes” [1].

Cagan also gave some practical guidance: - **Prototypes are for learning, not for shipping.** Use them to test ideas with customers, stakeholders, and engineers. Faster delivery still costs tokens [1]. - **PMs own business viability.** That covers channels, costs, runtime and monetization, compliance, legal, and privacy [1]. - **For AI products, evals are product discovery.** They need to continue across discovery and delivery [1]. Don't put a serious promise on the roadmap until you've done discovery on it, or switch to roadmaps built around outcomes [1].

Survey data from a sponsored post points the same way. Aakash Gupta sum-

marizes Atlassian’s *State of Product 2027*, a survey of 1,000 senior product professionals. 80% say AI helps them ship faster, yet customers don’t get value any sooner, and 69% say decision-making is as slow as before [2]. Only 24% say Product leads cross-functional work, down from 47% in 2025 [2]. Prioritization “still runs on opinion” because the inputs are spread across many tools [2]. In a separate note, Gupta says job postings at OpenAI, Anthropic, and Google DeepMind ask PMs to write evals and prototype with code. Meanwhile, the tickets-and-standups side of the job is being automated [3].

OpenAI’s Dots: trust controls are the product

A Mind the Product breakdown describes OpenAI’s Dots as an always-on agent. Each one runs on its own cloud computer and browser and looks for work without being asked [4]. In a launch demo, a Dot noticed a product change, flagged that the launch slides were out of date, and offered two revised versions. A human picked between them [4]. The presenter argues the controls are the real UX. Users decide whether each action runs automatically, needs approval, or is never allowed. There is also a status view, an automatic review of consequential actions, and a stop option [4]. A test question for teams building agents: “what’s the most consequential thing your agent can do without asking?” [4]

Implications for PMs: - **Agents will use your product.** Test whether they can get through login, onboarding, and modals. Learn to identify agent traffic, because it will distort session metrics [4]. - **Per-seat pricing may not fit.** Usage can grow while the number of human users shrinks [4]. - **New buying route.** Eligible US enterprises can apply existing OpenAI spending commitments to approved software from 32 marketplace partners [4]. - **Vendor questions.** OpenAI held back GPT 6.1 Astra for not staying within scope and authorization, and the launch included no error rates. Ask vendors how they test for this [4].

Charging agents for API access can backfire

On SaaStr’s *The Agents*, the hosts say their agents make 35,000–40,000 API calls a day. One estimate put the cost of that access at up to \$240,000 a year [5]. Their agent’s first suggestion was to call the API less and copy the system-of-record data into Postgres [5]. The bigger risk, they argue, is new deals: “I doubt an agent would recommend picking any system of record that materially charges for API access” [5].

Owning the work, not just retrieving it

On a16z, procurement startup LEO’s founder discussed a four-step ladder for agents: retrieval, process, policy (applying judgment), and principal (weighing broader trade-offs). Incumbents were described as offering mostly retrieval with “a little bit of process” [6]. LEO found that invoice matching was only about

20% of the work. The other 80% was exceptions such as fraud and mismatches, so it moved to handling those [6].

Practitioner notes

- **“Which customer?”** An r/ProductManagement thread on sales-driven requests reached consensus: a request is “a data point,” not a spec. Find out who asked and why, and talk to the customer directly [7]. Another commenter suggests waiting a couple of weeks to see whether the request comes up again, as a check on recency bias [8].
 - **Synthesis.** Teresa Torres notes that teams often skip deep synthesis after interviews and keep the two or three things they remember. “We’re competing with making shallow synthesis better” [9].
-

Sources

1. Fireside Chat with Marty Cagan and Elias Lieberich - ProductTank Berlin
2. substack
3. substack
4. Everything You Need To Know About OpenAI’s Dots in 17 minutes
5. Agent Pricing is Chaos. Here’s What We’re Seeing From the Buyer Side | The Agents #015
6. Why AI Is Reinventing How Businesses Buy Everything
7. r/ProductManagement comment by u/cpt_fwiffo
8. r/ProductManagement comment by u/twopunchbear
9. X post by @ttorres