

Evals Become the AI PM's Product Artifact

PM Daily Digest

2026-07-27

Evals Become the AI PM's Product Artifact

By PM Daily Digest • July 27, 2026

A practical guide to treating evals as product artifacts, using real model failures to build test loops, and planning for the capability changes AI introduces. It also covers a Claude eval example and the technical, hands-on expectations emerging for AI PMs.

Big Ideas

In AI products, the eval can become the central product artifact

At Anthropic, research PMs describe “**evals are the new PRDs**”: detailed user feedback becomes reproducible tests that represent a user need and measure whether a model version improves it. [1] This changes the PM job from specifying an interface alone to defining what a good model behavior looks like.

Why it matters: visible output can conceal weak judgment. AI may let people create artifacts across more role boundaries, but producing an artifact is not the same as possessing the expertise behind it. [2] PMs therefore need mechanisms—such as evals—to assess quality rather than relying on plausible-looking output.

Treat AI as a capability redesign, not a simple productivity upgrade

A useful lens maps nine possible movements of work: specialization, diffusion, centralization, integration, embedding, rebundling, externalization, elimination, and loss. [2] AI can spread codified expertise, embed knowledge in tools, and rebundle work into broader roles—but it can also create needs for evaluation, governance, orchestration, and exception handling. [2]

Apply it: when automating a workflow, ask not only what task disappears, but which routine activities currently develop the judgment people will need for difficult cases. If those activities vanish, identify how that judgment will be built and maintained. [2]

Tactical Playbook

Turn model failures into a working eval loop

1. **Collect the precise failure, not a summary.** Ask for the exact user request, model response, and situation in which the behavior failed. [1]
2. **Read the trajectory.** Inspect transcripts closely enough to distinguish, for example, hallucination from overconfidence; the theme of a failure can be nuanced. [1]
3. **Convert repeated failures into test cases.** Define an expected or “golden” answer and build a set of examples that consistently represents the pain point. [1]
4. **Run the set on each version.** Add the cases to the eval repository and use them to check new model versions. [1]
5. **Keep PRDs where they add value.** Use them for broad stakeholder alignment and ambiguous, early-stage opportunities where product vision must be explored; use evals as the shorthand for a defined model-improvement loop. [1]

Why it matters: an offline eval passing while users remain unhappy is a signal to revisit whether the eval captures the actual user problem—not merely to declare the feature complete. [3]

Case Studies & Lessons

Claude’s JSON failures became an eval set

Early Claude feedback that it was poor at following instructions was investigated at the level of individual prompts and responses. The team found that roughly **80%** of the reported issue was failure to produce the right JSON, then created an initial set of **30–40 examples** as an eval. [1]

Schema-following later became fundamental to Claude’s ability to act as an agent, since structured output supports API access and tool calls. [1]

Lesson: broad feedback such as “it doesn’t follow instructions” is not yet a roadmap item. Product work begins by finding the specific, recurring behavior that can be evaluated and improved.

Career Corner

Prepare for technical depth—and stay hands-on

Nvidia, OpenAI, and Anthropic reportedly use technical PM interview questions, with an expectation that PMs can earn the respect of engineering and research counterparts. [3] Key preparation areas include Transformers and attention; agents, MCP, and APIs; routing and unit economics; RAG; system trade-offs; and evaluation. [3]

Apply it: practice concise explanations, then connect each technical decision to a product consequence—for example, explain how routing a share of traffic to a cheaper model affects margins at scale. [3]

For leaders, technical fluency is not presented as a delegation-only skill: Anthropic’s product leadership emphasizes personally shipping with models and staying close to the details. [1] Pair that with an inward-facing career habit: aim to deliver strong work even under an average or absentee manager, rather than only seeking ideal management conditions. [4]

Sources

1. How Anthropic builds products like Claude Code before the AI models are ready | Dianne Penn
2. TBM 432: Bundling & Unbundling Capabilities (and AI)
3. substack
4. The Antithesis Principle