

# Managing Agents, Making Better Calls, and Leading with Availability

PM Daily Digest

2026-03-30

## Managing Agents, Making Better Calls, and Leading with Availability

*By PM Daily Digest • March 30, 2026*

This issue centers on three shifts PMs should internalize: managing agents like teammates, making faster decisions with incomplete data, and treating availability as a real leadership framework. It also includes execution templates, concrete agent case studies, and career skills to build now.

### Big Ideas

#### 1) PMs need an agent-management operating model

“2005: growth teams optimized funnels / 2015: growth teams optimized loops / 2026: growth teams optimize agents” [1]

Claire Vo and Julie Zhuo describe the same shift from different angles. Agents work best when PMs use classic management skills: define purpose clearly, scope a role, give the right context and tools, onboard progressively, and split work across specialized agents when context gets too large. Humans still need to define what great looks like and curate the final outcome [2, 3].

**Why it matters:** PM leverage is moving beyond prompting. The harder questions are who does what, with what permissions, in what context, and how success is judged [2, 4].

**How to apply:** - Define success before setup: spell out the outcome, and what an excellent vs. mediocre result looks like [3]. - Start with narrow roles and limited access, then expand trust over time—for example, calendar access before email drafting or sending [4]. - Use multiple specialized agents instead of one overloaded generalist when work spans different contexts [2]. - Treat onboarding as product design: make it simple, conversational, and oriented around helping the user feel like a winner [2].

## 2) Better product decisions come from higher understanding, not perfect certainty

Julie Zhuo describes Meta’s product culture as hypothesis-led and experiment-based: use data to test whether a belief is true, and change course if it is not. She defines a high-quality decision by whether it drives the intended outcome over time, and by whether the process was rigorous enough to trust in the short term. Her framing at Sundial is similar: if AI can raise understanding from roughly 30% to 50-60%, teams can make better, faster decisions without waiting for perfect information [3].

**Why it matters:** PMs do not need certainty to move; they need clearer hypotheses, better validation, and enough confidence to act before the window closes [3].

**How to apply:** - Start with the outcome you want, not the feature you want to build [3]. - List the assumptions behind the decision and validate them with customer conversations or data [3]. - Compare ideas by building and testing them, especially in small teams, instead of resolving everything in alignment meetings [3]. - Make the call once understanding is materially better, even if it is still incomplete [3].

## 3) Availability is a leadership framework, not a personality trait

Availability is framed here as being reachable, present, engaged, and reliable in follow-through—and as a common trait among the best leaders, even though it rarely appears in standard leadership frameworks [5]. The operating behaviors are simple: make time, make space, make calls, and make good on commitments [5].

**Why it matters:** Chronic unavailability does more than slow execution. Teams stop escalating issues, trust erodes through unclosed loops, and people start proposing smaller, safer bets [5].

**How to apply:** - Protect calendar space for decisions and people who actually need your input [5]. - In meetings, reduce context-switching and be fully present [5]. - Decide when the decision is needed, rather than deferring to another meeting or more data by default [5]. - Audit your own follow-through, and ask specific questions such as “What took too long from me?” or “When did you wish I’d been around?” [5].

## Tactical Playbook

### 1) Set up an agent like a new teammate

1. Define the purpose: what outcome matters, and what excellent vs. mediocre performance looks like [3].
2. Scope one role at a time, with the right tools, context, and style guidance [2, 3].

3. Start with progressive trust instead of full autonomy on day one [4].
4. Learn the agent’s strengths and weaknesses through repeated interactions, just as you would with a new teammate [3].
5. Split work into specialized agents when context gets crowded [2].

**Why it matters:** This turns vague AI experimentation into a repeatable operating model that matches how managers already create clarity and accountability [2].

## 2) Raise decision quality without slowing the team down

1. Write the outcome you want in plain language [3].
2. Note the assumptions that must be true for that outcome to happen [3].
3. Validate those assumptions quickly with users, data, or both [3].
4. Stress-test the downside and likely misinterpretations before shipping [3].
5. Build competing ideas in parallel when possible, then judge them on evidence rather than debate [3].
6. Decide when you have enough understanding to move, not when uncertainty disappears [3].

**Why it matters:** The goal is not perfect process. It is a process rigorous enough to trust, while still moving before timing or relevance disappears [3].

## 3) Run weekly execution reviews around outcomes, not activity

1. Require each weekly update to include only three lines: **Decision**, **Owner**, and **Definition of Done** [6].
2. Clarify who is **Leader Accountable**, **Responsible**, **Consult**, and **Inform**, then pair that with **Objectives**, **Key Results**, and **Actions** [7].
3. At the end of the week, review whether the objective was achieved and whether the key result proved it—not whether a list of actions was completed [7, 6].
4. Keep ownership and decisions visible as work evolves, using systems like GitHub and Slack if that is where the work already lives [8].

**Why it matters:** This is a lightweight way to reduce the common drift between code merged and customer outcome shipped in small teams [6].

## 4) When driving change, show more than you tell

1. Start by asking what you actually want: validation, change, or both [9].
2. Assume challenged beliefs will trigger defensiveness on both sides; that is a human reaction, not proof of bad intent [9].
3. Avoid treating workplace relationships like deep personal bonds; many are situational and will not satisfy a deeper need to feel fully understood [9].
4. Use evidence and working examples to create movement, instead of relying on abstract comparisons or logic alone [9].

**Why it matters:** PMs often carry the burden of proof when proposing a better way of working. Showing a better path can be more effective than trying to win the argument outright [9].

## Case Studies & Lessons

### 1) ChatPRD’s sales agent turned manual founder work into a daily loop

Claire Vo’s agent *Sam* runs a daily PLG sweep across new signups, checks for company domains, uses Exa people search to find likely decision-makers, sends soft personalized outreach, routes some accounts back to Claire, handles international accounts differently, cleans up the CRM, flags stale deals, drafts customer emails, and runs QBRs [2]. She says it replaced a part-time salesperson who had been doing the work 10 hours per week, and she highlights that the setup is highly tunable as the workflow changes [2, 4].

**Why it matters:** This is a concrete example of an agent creating real economic value in a PM-adjacent growth workflow [2, 4].

**How to apply:** Pick one recurring revenue or ops workflow with clear inputs, routing rules, and review points before expanding scope [2, 4].

### 2) Agent support made a new course viable before extra hiring

Vo says she and her co-instructor built an executive course in Claude code, then added *Sage* to project-manage launch prep. Sage tracks the launch timeline, nudges them to post on LinkedIn, ingests research via API, stores it in the repo, and decides where it belongs in the syllabus. She says this let them spin up the first version without hiring an ops person, content manager, or software engineer [2].

**Why it matters:** Small teams can use agents to stand up real operating capacity before they can justify full-time hires [2].

**How to apply:** Start with coordination-heavy work—timelines, reminders, content filing, and draft generation—where the value of consistency is easy to see [2].

### 3) Meta did not ship the feature users asked for most

Julie Zhuo described the post-like-button demand for a dislike button as one of the top user requests. The team still rejected it after thinking through best- and worst-case outcomes and how the feature could be misinterpreted. They eventually pivoted to reactions, which felt more neutral and expressive [3].

**Why it matters:** Strong demand is not enough. PMs still need to test whether a feature advances the product’s intent and what negative interpretations it might introduce [3].

**How to apply:** When a request is popular, pair request volume with an explicit downside review before prioritizing it [3].

## Career Corner

### 1) Management skill is becoming product leverage again

Claire Vo’s view is direct: management skills matter more than technical skills for getting agents to work well. Her argument is that leaders already know how to onboard, role-scope, and set people up for success; those same skills transfer to agent systems [2, 4]. Julie Zhuo’s **Purpose / Process / People** framework points to the same muscle: define success, define how the work gets done, then learn the strengths and weaknesses of the humans or agents doing it [3].

**Why it matters:** This is a durable skill set that applies across product leadership, AI delegation, and team design [2, 3].

**How to build it:** Start with one internal workflow and practice writing clearer success criteria, tighter role scopes, and better review loops [3, 2].

### 2) Senior PMs may need to get more hands-on, not less

Zhuo says she now asks whether she really needs to interrupt another human, or whether she can use the available tools and do it herself. She also expects future teams to be much smaller, with fewer alignment meetings and more execution per person [3].

**Why it matters:** Smaller teams raise the execution bar for every individual contributor and leader [3].

**How to build it:** Prototype, analyze, or instrument the first version yourself before turning it into a staffed project [3].

### 3) Durable careers are built on problems, not just solutions

“fall in love with a problem, not a solution.” [3]

Zhuo argues that the most durable work is anchored in universal problems, while solutions should be treated as hypotheses to validate quickly with users or buyers [3].

**Why it matters:** Problems outlast individual tools, interfaces, and product shapes [3].

**How to build it:** When describing your work, lead with the underlying user or business problem you are pursuing, then show how you tested candidate solutions [3].

#### 4) If you want more influence, audit your availability

When leaders repeatedly fail to close loops, teams stop bringing them issues and stop expecting useful follow-through [5]. Over time, that also shrinks the team's ambition because people learn to avoid bold ideas that need leader support [5].

**Why it matters:** Influence decays quickly when the organization learns to route around you [5].

**How to build it:** Review a recent week's commitments and ask your team where your response time or follow-through slowed them down [5].

### Tools & Resources

#### 1) OpenClaw

Vo describes OpenClaw as a strong reference point for agent product design: easy onboarding, self-learning and self-improving behavior, and open-source visibility into how the system works [2].

**Why explore it:** It is both a tool and a live product example for PMs learning how agent experiences should be structured [2].

**How to use it:** Inspect the docs or code to study onboarding, task scheduling, and other design choices you may want to copy into your own agent workflows [2].

#### 2) Sundial

Sundial's premise is to productize analytics expertise with AI so decision-makers can move from roughly 30% understanding to 50-60% understanding and make better, faster decisions [3].

**Why explore it:** It is a concrete example of an AI product aimed at decision quality rather than just report generation [3].

**How to use it:** Consider it when your bottleneck is understanding ambiguous data fast enough to support a product or business decision [3].

#### 3) The weekly Decision / Owner / Definition of Done template

A lightweight weekly update format—paired with LARCI and OKRA—forces teams to define ownership, success, and review against outcomes instead of status prose [6, 7].

**Why explore it:** It is simple enough to adopt immediately in startup or small-team execution rhythms [6].

**How to use it:** Make it the default weekly update format, then review the output against objectives and key results at week's end [6, 7].

#### 4) Julie Zhuo's 3Ps framework

**Purpose, Process, and People** provide a compact rubric for both team management and agent management: what success is, how the work should happen, and who or what is best suited for it [3].

**Why explore it:** It gives PMs one reusable frame for people leadership, AI delegation, and execution design [3].

**How to use it:** Run every new project, workflow, or agent setup through the same three questions: what outcome matters, how should the work happen, and who or what should do it [3].

---

#### Sources

1. X post by @andrewchen
2. From skeptic to true believer: How OpenClaw changed my life | Claire Vo
3. Dean's Speaker Series | Julie Zhuo | Co-Founder & CEO, Sundial; Author
4. X post by @lennysan
5. The 12th of Never
6. r/startups post by u/HiSimpy
7. r/startups comment by u/edkang99
8. r/startups comment by u/HiSimpy
9. TBM 413: In That Space Is Our Power