# Continuous Intelligence
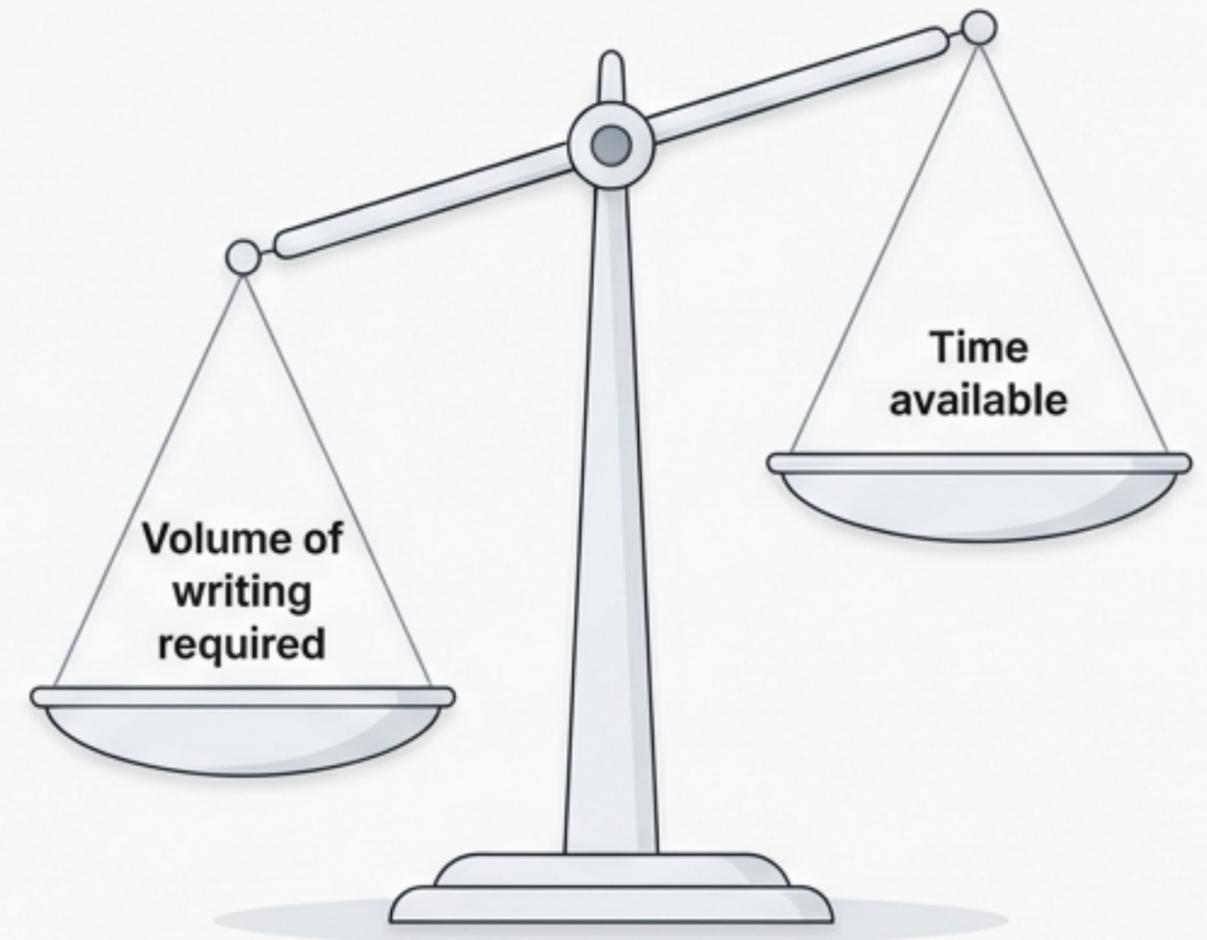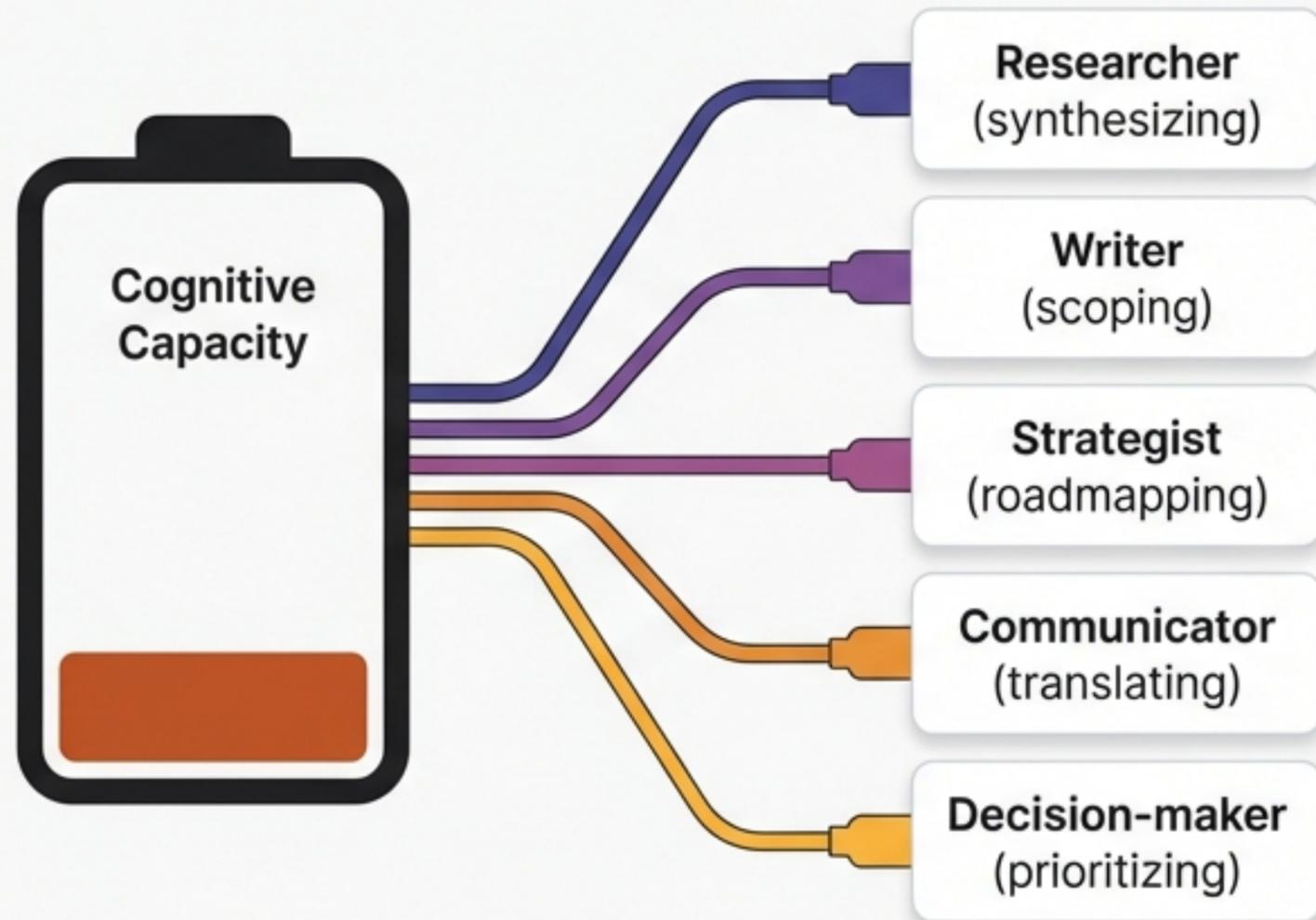
The PM's Editorial Operating System for AI-Augmented Workflows
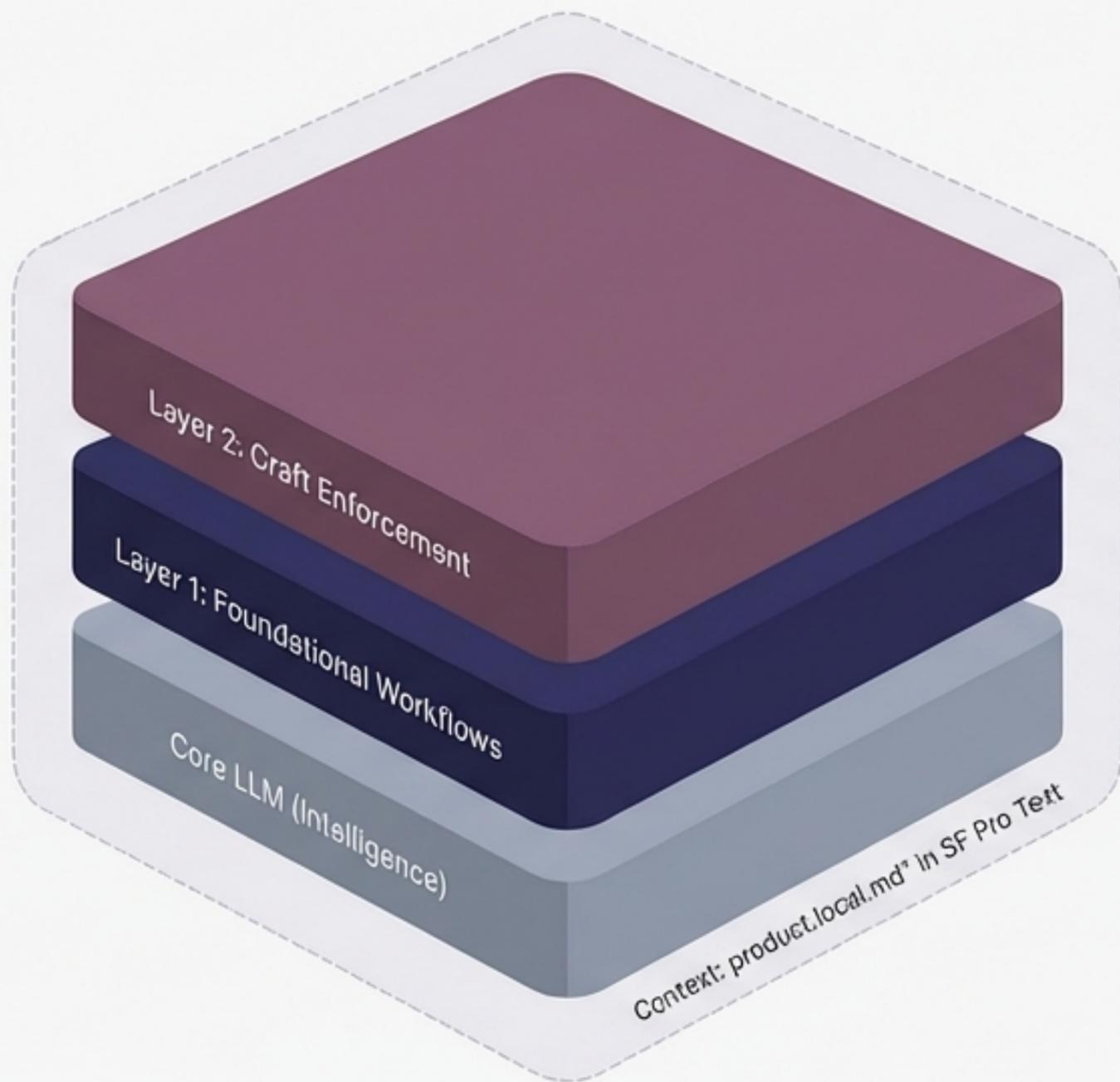
> System initialized. Closing the document gap.

# The Document Gap, Not a Skill Gap

Cognitive Capacity

- Researcher (synthesizing)
- Writer (scoping)
- Strategist (roadmapping)
- Communicator (translating)
- Decision-maker (prioritizing)

Volume of writing required

Time available

Most PMs know how to **write a good spec**. They just cannot write it at the quality it deserves **AND do everything else** simultaneously.

NotebookLM

# The Two-Plugin Architecture



Layer 2: Craft Enforcement
Layer 1: Foundational Workflows
Core LLM (Intelligence)

Context: product.local.md" in SF Pro Text

## Command Routing

### Layer 1 (Official Plugin)
Deep Indigo

/write-spec

/roadmap-update

/stakeholder-update

### Layer 2 (Custom Plugin)
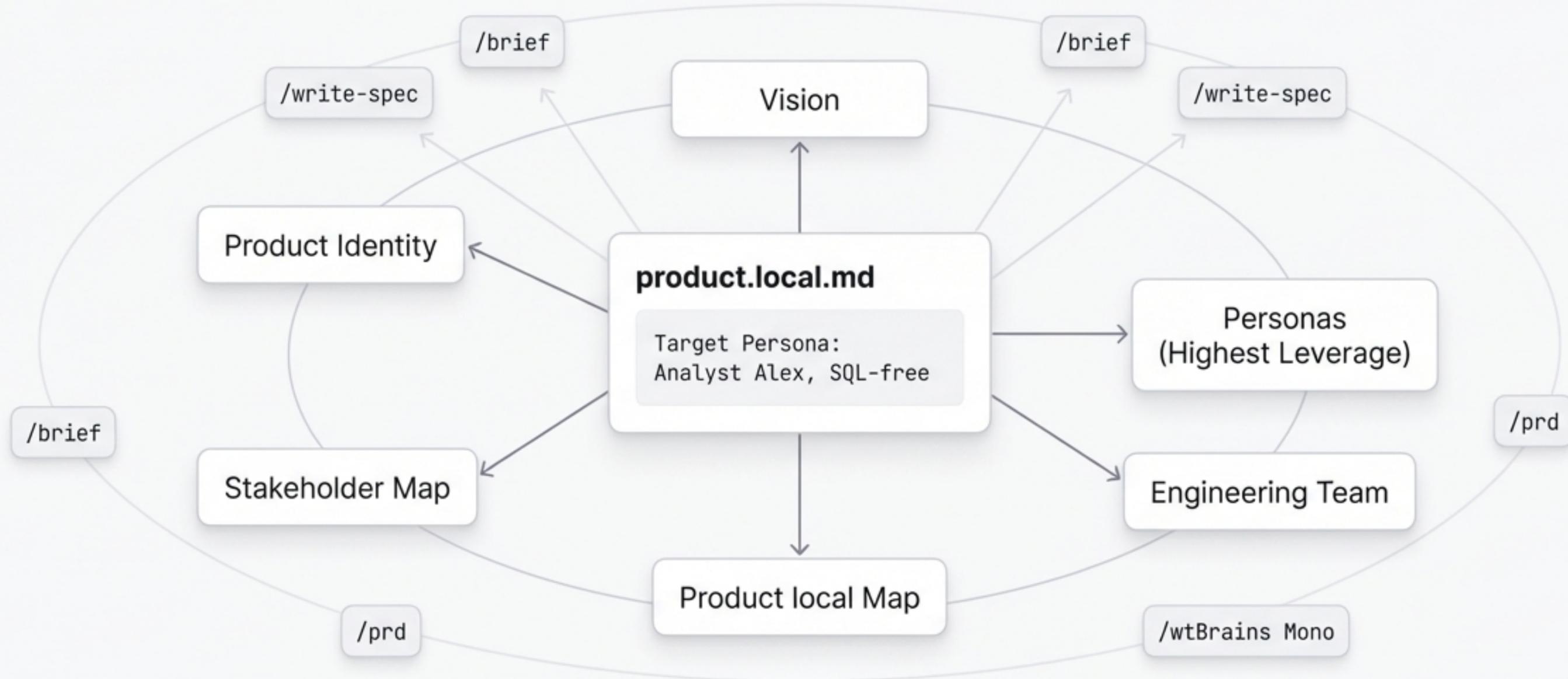Muted Plum

/brief

/prd

/stories

NotebookLM

# Context as the Gravitational Core



Vision

Product Identity

**product.local.md**

Target Persona:
Analyst Alex, SQL-free

Personas
(Highest Leverage)

Stakeholder Map

Engineering Team

Product local Map

/brief
/write-spec
/brief
/write-spec
/brief
/prd
/prd
/wtBrains Mono

A /brief command without this core produces generic templates. With it, it produces a document written by a two-year team veteran.

# Discovery: Framing the Right Problem

## Solution-Prescriptive ✗

Uses verbs like build, add, integrate.

Anchoring risk is high.

Leads to confirmation bias.

> "Add a dashboard."

## Problem-Focused ✓

Uses verbs like struggle, wait, lose.

Anchoring risk is low.

Leads to evidence-based discovery.

> "Analysts cannot automate reports."

### Command Execution

```
> /brief
Analyzing input...
Enforcing problem-focus...
Generating Discovery Brief...
```

**Rule:** NEVER propose a solution in a problem brief. Acknowledging uncertainty is strength.

# The Spec Quality Spectrum

## The Vague Spec

"Users should manage notifications."
JetBrains Mono

Engineers make product decisions.
SF Pro Text

Rework.
SF Pro Text

## The Craft Middle Ground

Complete on what matters, silent on implementation.
SF Pro Text

- Explicit scope boundary (IN/OUT OF SCOPE).
SF Pro Text

## The Over-Specified Spec

50 pages of micro-interactions.
SF Pro Text

PM does engineering.
SF Pro Text

Resentment & rigidity.
SF Pro Text

```
> /write-spec
Translates discovery evidence
into sprint-ready architecture.
```

NotebookLM

# The Acceptance Criteria Filter
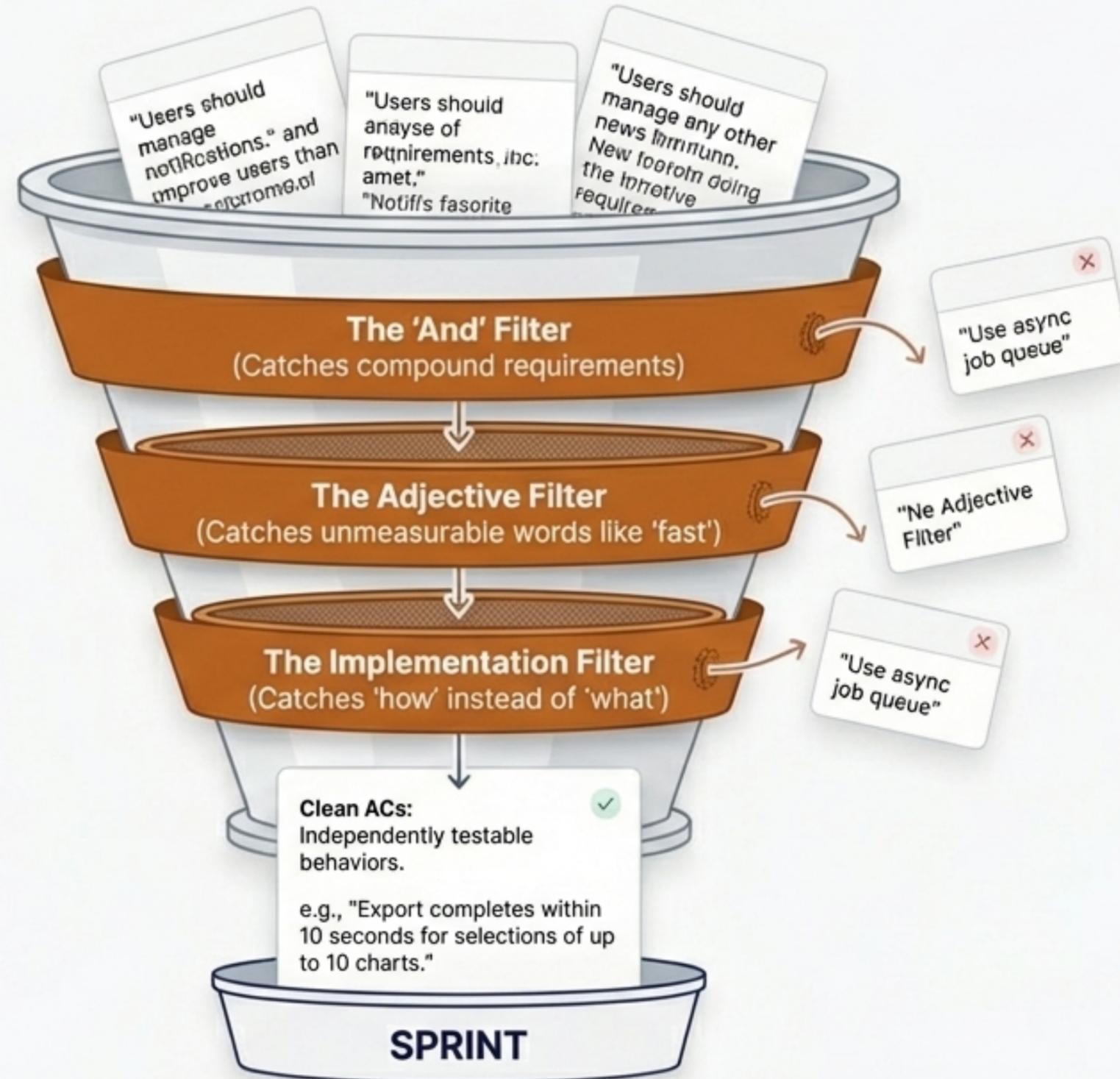
# PRDs & The 60% MUST Rule
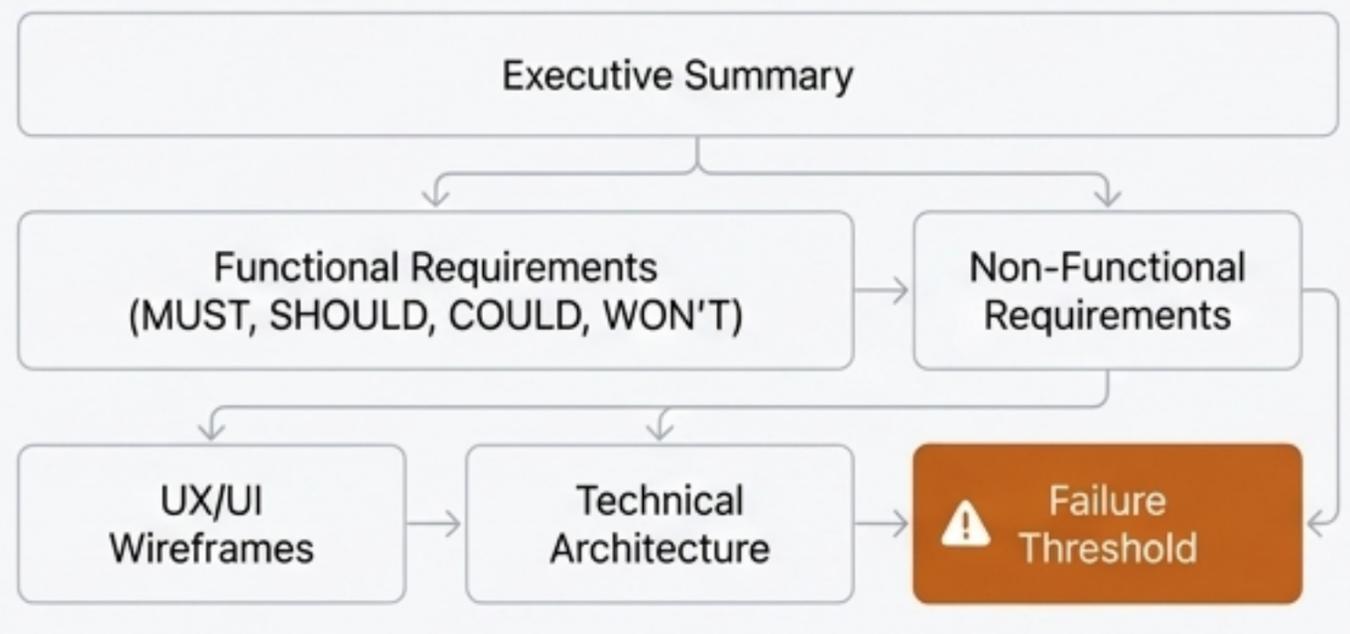
50%

**60%**

25%        75%

0%

100%

If more than 60% of functional requirements are labeled MUST, the scope is too large.

The MUST Test: If this were not ready on launch day, would we delay the entire initiative?
If no, it is a SHOULD.

```
> /prd
Loading PRD architecture...
Analyzing requirements...
```

**Architecture Map**

Executive Summary

Functional Requirements
(MUST, SHOULD, COULD, WON'T)

Non-Functional
Requirements

UX/UI
Wireframes

Technical
Architecture

⚠ Failure
Threshold

**Key Insight:** A PRD requires a Failure Threshold. Define the specific outcome that forces a change of direction before launch to prevent sunk-cost escalation.

# Execution: Story Splitting Triggers

```
Evaluate User Story
```

- **>7 Acceptance Criteria?** → Split by User Flow
- **Multiple Personas?** → Split by Persona
- **Exceeds Sprint Capacity?** → Split by Deliverable Phase
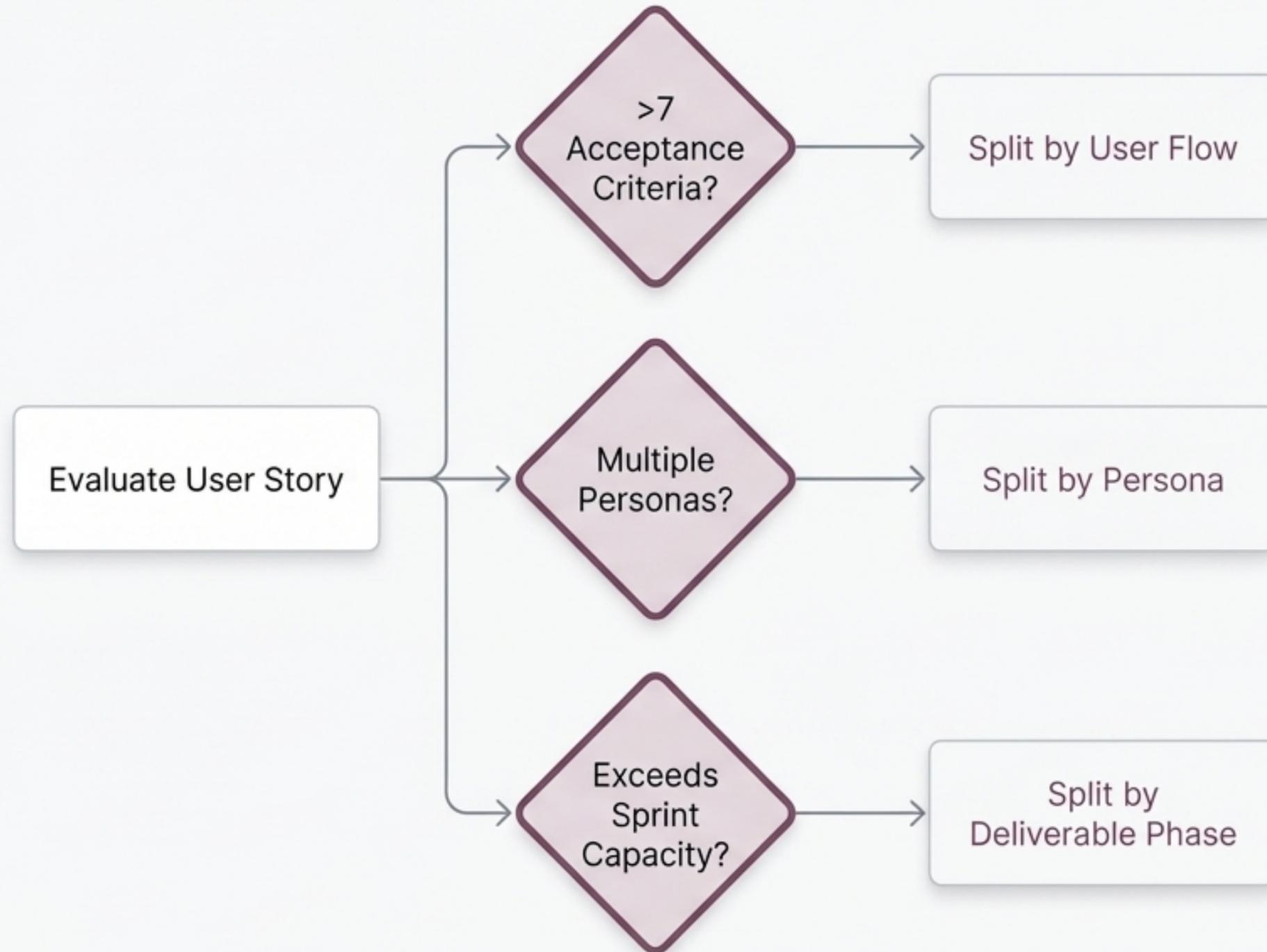
```
> /stories
```

## Mandatory Anatomy

- ✓ Persona (from product.local.md)
- ✓ Capability (NOT a UI element)
- ✓ User Outcome (NOT a system action)

# Roadmap Frameworks & Audience

## Now / Next / Later (Default)
Best for external comms/execs. Avoids false date expectations.

## Quarterly Themes
Best for strategy alignment. Use for internal planning.

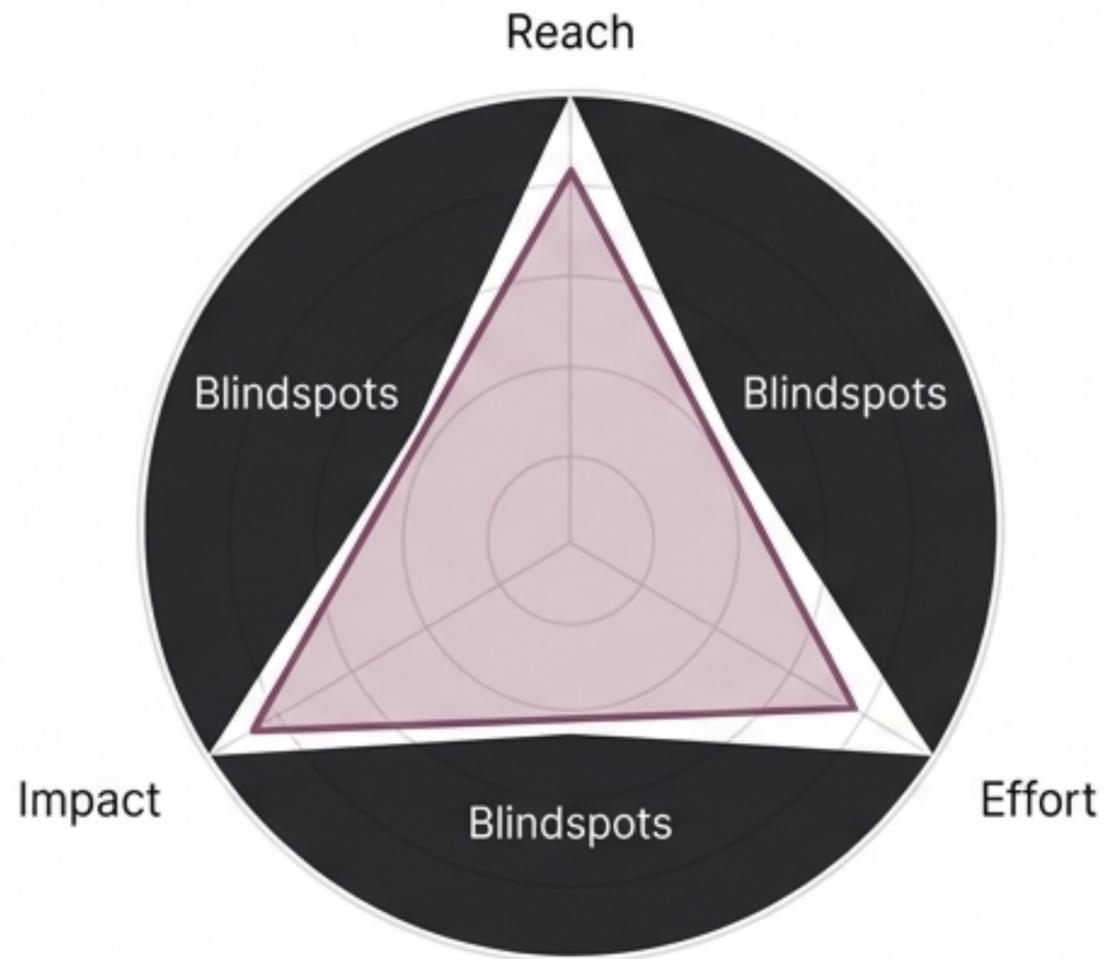## OKR-Aligned
Best for executive accountability.

## Timeline / Gantt
Best for internal engineering ONLY.
Avoid for customers (reads as a contract).

> `/roadmap-update`   Generates tailored roadmap outputs based on audience parameters to prevent false precision traps.

NotebookLM

# Prioritization: RICE Blindspots

## RICE Framework & Blindspots



> /prioritise

### Judgment Gate

## 1. Strategic Necessity (The Override Test)

If a low-scorer will be built anyway (e.g. CEO promise), document the override. Don't fake the math.

### Judgment Gate

## 2. Data Gaps

Low confidence + high effort = Trigger a 2-week Discovery Spike.

### Judgment Gate

## 3. User Love (The Regret Test)

If we shipped nothing else, what would our best customers be most grateful for?

NotebookLM

# Sprint Planning: Capacity Arithmetic

$$[\text{Raw Days}] - [\text{PTO} + \text{On-Call} + \text{Meetings}]$$
$$\times\ 75\% = [\text{Planning Capacity}]$$

### Why 75%?

Interrupts and complexity always consume the remaining 25%. 100% capacity guarantees missed commitments.
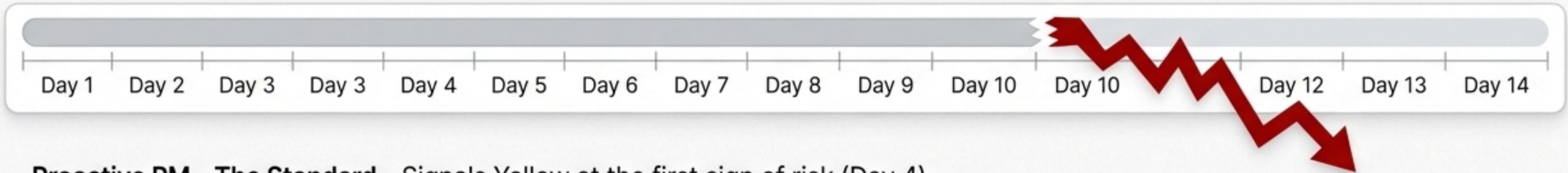
### Units

Measured in Person-Sprints.

> /sprint-planning    Focus on a singular, testable Sprint Goal bounded by undeniable mathematics.

# Proactive Operation & Risk Signaling

## Sprint Timeline

**Reactive PM** - Stays Green until Day 10, then drops suddenly to Red (Chaos/Escalation).

| Day 1 | Day 2 | Day 3 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 10 | Day 12 | Day 13 | Day 14 |

**Proactive PM - The Standard** - Signals Yellow at the first sign of risk (Day 4).

| Day 1 | Day 2 | Day 3 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 | Day 10 | Day 11 | Day 12 | Day 13 | Day 14 |
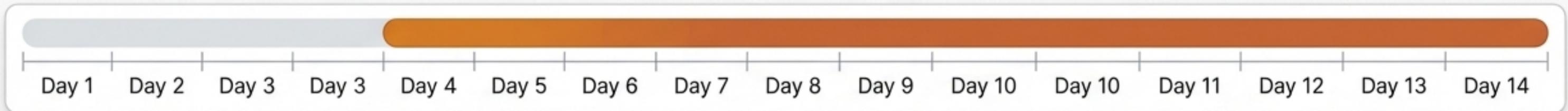
## ROAM Risk Framework

[Resolved]   [Owned]   [Accepted]   [Mitigated]   ⚠ If an 'Owned' risk lacks a named owner and mitigation plan, it is an unmanaged risk.

NotebookLM

# The Stakeholder Translation Matrix

**Sprint 1 Status**

## Exec

Sprint goal at risk due to backend complexity.

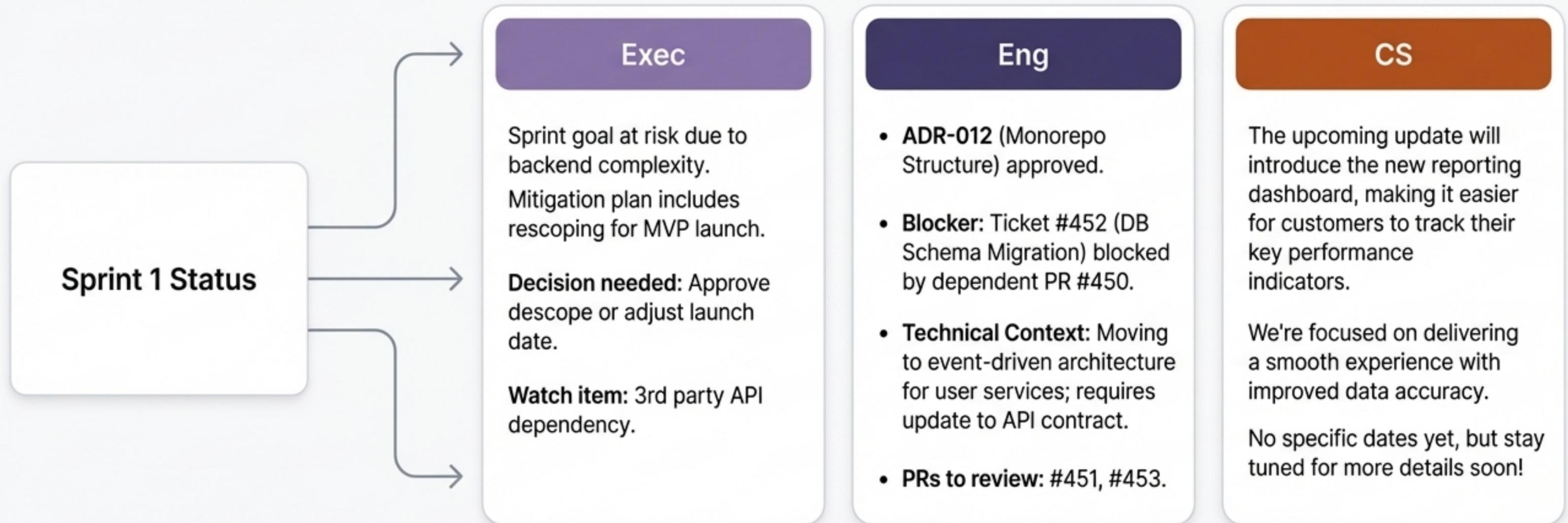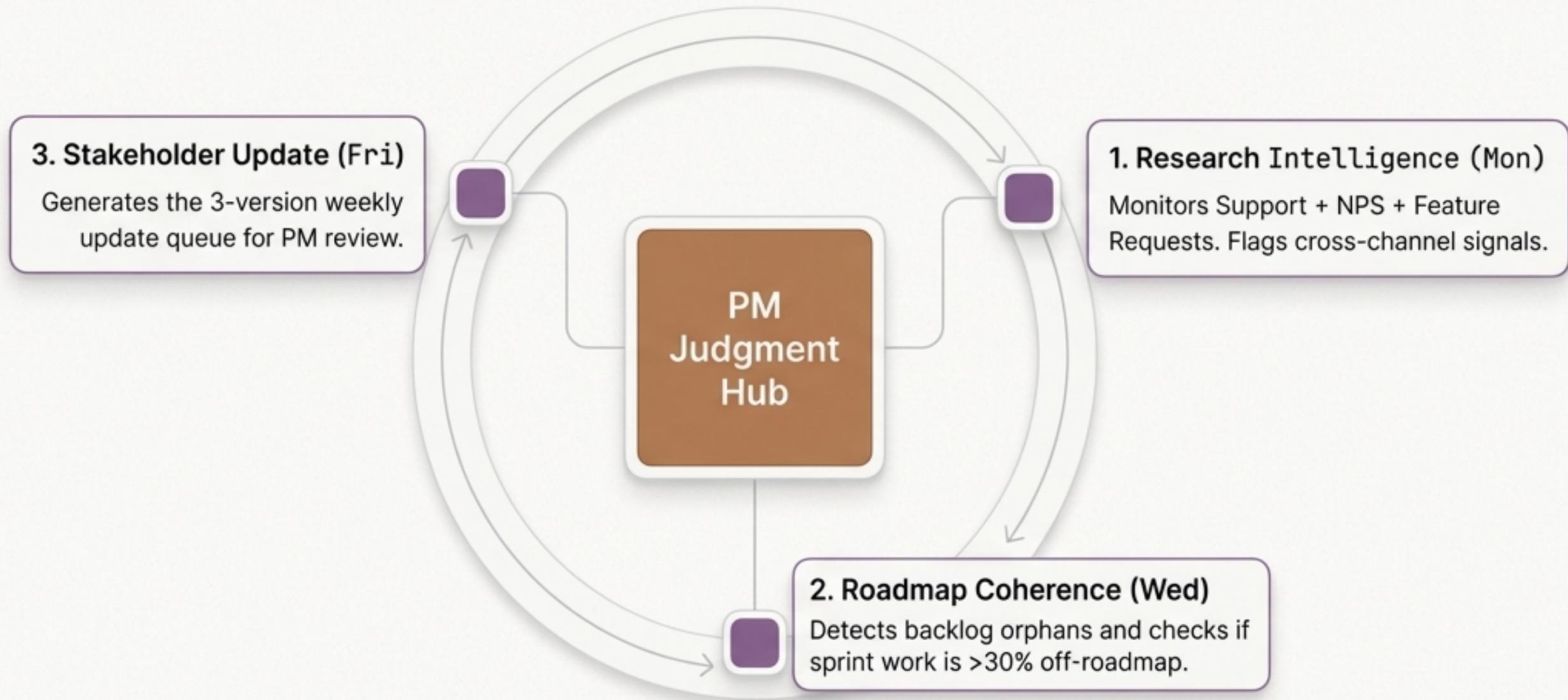Mitigation plan includes rescoping for MVP launch.

**Decision needed:** Approve descope or adjust launch date.

**Watch item:** 3rd party API dependency.

## Eng

- **ADR-012** (Monorepo Structure) approved.

- **Blocker:** Ticket #452 (DB Schema Migration) blocked by dependent PR #450.

- **Technical Context:** Moving to event-driven architecture for user services; requires update to API contract.

- **PRs to review:** #451, #453.

## CS

The upcoming update will introduce the new reporting dashboard, making it easier for customers to track their key performance indicators.

We're focused on delivering a smooth experience with improved data accuracy.

No specific dates yet, but stay tuned for more details soon!

```
> /stakeholder-update
```
Core Rule: Never send the same version of an update to different audiences.

NotebookLM

# The Continuous Intelligence Engine

**3. Stakeholder Update (Fri)**
Generates the 3-version weekly update queue for PM review.

**1. Research Intelligence (Mon)**
Monitors Support + NPS + Feature Requests. Flags cross-channel signals.

PM Judgment Hub

**2. Roadmap Coherence (Wed)**
Detects backlog orphans and checks if sprint work is >30% off-roadmap.

Agents do not replace PM judgment. They remove the friction between the judgment and the document that communicates it.