



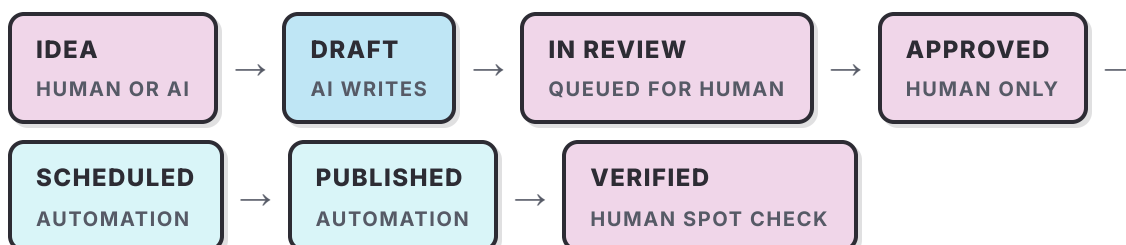
The Approval Pipeline: *AI drafts at scale, a human ships every post.*

The status flow I run to publish across multiple brands without ever letting automation speak for a human unreviewed. By Debbie Moynihan.

TL;DR

Give every piece of content one pipeline with seven statuses. AI and automation own the mechanical moves. Exactly one transition, into Approved, belongs to a human, and that gate is designed like a system component: an owner, a recurring time slot, a batch format, and a definition of done. This is how you get AI volume with human judgment on every single piece that ships.

THE STATUS FLOW, AT A GLANCE



☐ AI does the work ☐ human judgment ☐ automation moves it

 *the only human-owned move*

One rule keeps this safe, and we keep it because of our own launch, not on principle: automation only picks up content in Approved, and only a human puts content there. Everything else can be delegated without risk, because nothing reaches the public without passing that gate.

WHO TOUCHES WHAT

AI employees

Generate drafts into the calendar, in the brand voice, at whatever volume the plan calls for. They never touch a status past In Review.

The human approver

One named person per brand feed. Their only pipeline power is the one that matters: moving a piece to Approved, back to Draft with a note, or to killed.

Automation

Watches for Approved, schedules into the posting slots, publishes, and records the result. It has no opinions and needs none.

The verifier

A human eyeball on the live feeds on a weekly cadence, confirming the published feed matches the calendar. Output is the metric, not system health.

Build it in five steps

1 Make one queue the single source of truth

One content calendar, one row per piece, one status column. If a post is not in the queue, it does not exist. Multiple lists are how pieces get lost and double-published.

2 Define the seven statuses and their owners

Copy the flow above. Write down, next to each status, who or what is allowed to move a piece out of it. Ambiguity here becomes a stall later.

3 Design the human gate like software

The approval step gets an owner, a recurring calendar block sized to the queue, and a batch format: one stack, one decision per piece, approve or reject or edit-and-approve. Approval that happens whenever there is a spare moment happens never.

4 Write the definition of approved

Approved means good enough to publish under your name, not perfect. List your blocking flaws in advance: wrong facts, wrong voice, wrong offer. Everything else ships. A minor typo in a personal-voice post is not a blocking flaw; decide that once, in writing, instead of re-litigating it per post.

5 Watch queue depth, not system health

Check how many pieces sit in each status weekly. A growing pile in In Review is your bottleneck announcing itself. A quiet published feed is an alarm, and it is the quietest one you will ever get.

FROM THE FIELD

We learned step three the honest way. Our engine ran flawlessly for weeks and published almost nothing, because every mechanical step was specified in detail and the approval step got one word in the plan: review. No owner on the calendar, no time slot, no definition of done. The automation was never the problem. The moment we designed the gate, the queue started draining.

EDGE CASES THE PIPELINE MUST ANSWER

a Edit after approve

Any edit to a piece past Approved sends it back to In Review. No exceptions. An edited post is a new post; the gate covers what actually ships, not what was once read.

b The failure path

A Scheduled item that fails to publish rolls back to Approved and raises an alert. It never retries silently forever; a stuck retry loop is just the starving engine wearing a different hat.

c Multi-channel fan-out

One row per platform variant. Approval covers the variant, not the idea; the caption that works on one channel can be wrong on another, so each rendered version passes the gate on its own.

Regulated team? Slot a second gate, legal or brand, between In Review and Approved. The invariant does not change; the Approved door simply has two keys.

SOLO OPERATOR VERSION

Running this alone? Keep the invariant, collapse the ceremony. Four statuses are enough: Draft, In Review, Approved, Published. You are both writer-of-record and approver, and the review step becomes a standing ten minute self-review appointment. Keep the calendar block anyway. The appointment is what makes the gate real, not the org chart.

COPY-PASTE: THE STATUS SET

PASTE INTO YOUR PROJECT TOOL'S STATUS FIELD

IDEA	owner: anyone (human or AI)
DRAFT	owner: AI employee, writes in brand voice
IN REVIEW	owner: approver's queue, batch reviewed on a schedule
APPROVED	owner: HUMAN ONLY, definition of done applies
SCHEDULED	owner: automation, assigns publish slot

PUBLISHED	owner: automation, records post URL
VERIFIED	owner: human, weekly live-feed spot check

Rename freely; keep the two invariants. Only a human sets APPROVED. Automation only reads from APPROVED onward.

 *pin this next to your calendar block*

COMMON MISTAKES

- ✗ Specifying the software in detail and leaving the human step as a vibe.
- ✗ Letting review mean rewrite, so every approval costs thirty minutes of judgment and ego.
- ✗ Measuring whether the automation runs instead of whether work exits the system.
- ✗ Adding more AI drafting upstream of a starved gate, which just deepens the queue.
- ✗ Giving several people approval power on one feed, which means nobody owns the queue.

The gate is not bureaucracy. The gate is the product: your judgment, applied at scale.

PRO TIP

Size the approval block to the queue, not the other way around. If the weekly plan is twenty posts and your reviewer can decide three per minute in a good batch UI, that is a ten minute standing appointment. Protect it like a client meeting.

This is Field Notes No. 001, the first entry in the series. The pipeline above is the one I actually run across multiple brand feeds.