LLM Workflow Release Ops

Phil Stevens
philipstevens.github.io

Operate releases with eval gating, drift detection, and rollback discipline so you ship safely over time.

SCOPE	One or more production LLM workflows requiring ongoing release discipline and drift management.
PRIMARY OUTPUT	Gated releases + drift monitoring + threshold maintenance + rollback readiness (sustained over time).

Best Fit

- Your workflow is live, and regressions or drift are a recurring risk.
- You want repeatable releases with clear evidence and fast rollback.
- You already have (or want to establish) an eval harness and baseline gates.

Inputs Required

INPUT	MINIMUM	WHY IT MATTERS
Baseline eval suite	Runnable gate suite with must-pass subset	Release ops is about preventing regressions over time.
Telemetry access	Version identifiers in trace/logs; production signals	Without this, incident response becomes guesswork.
Release process	Canary/shadow/ramp capability (even manual)	Allows safe rollout and rapid rollback.
Owner plus cadence	Someone who can approve releases and review weekly	Ensures accountability and keeps drift monitoring actionable.

If you don't have these yet, the first month often starts with a short Build & Harden sprint to establish them.

How the Work Runs

CADENCE	WHAT HAPPENS
WEEKLY	Review production signals, sample quality on a rotating set, triage drift indicators and prioritize fixes.
PER RELEASE	Run eval gates on proposed changes, analyze deltas vs baseline, define rollout plan and rollback triggers.
MONTHLY	Update test sets from incidents, review and ratchet gate thresholds, calibrate alerts to reduce noise.

What breaks most release processes

- Gates that are too slow to run, so they're skipped
- Thresholds that never get ratcheted
- Drift signals without an owner
- Rollbacks that exist on paper only

Deliverables

ARTIFACT	PURPOSE
Maintained eval suite (test sets updated from incidents)	Keeps coverage current as product and failure modes evolve so yesterday's bugs don't recur tomorrow
Release gate reports	Documents what changed, what passed/failed, and ship decision so every release has a paper trail
Weekly stability memos	Summarizes trends, drift signals, and recommended fixes so problems surface before users report them
Drift monitoring dashboards and alerts	Surfaces regressions before users report them
Updated thresholds and gate configurations	Ratchets quality bar as workflow stabilizes
Rollback runbook (tested and current)	Ensures fast recovery when releases fail

Definition of Done

- ✓ Every release has a gate report documenting what changed, what passed/failed, and a signed-off ship decision.
- ✓ Drift and regression monitoring is in place with defined signals, owners, and documented response steps.
- ✓ Thresholds and test sets are maintained with versioned updates and review sign-off on each change.
- ✓ Rollback is documented and exercised at least once per quarter so recovery time stays under SLA.

Boundaries

- Building brand-new workflows from scratch (handled via Audit + Build & Harden).
- Large re-architecture or platform build work.
- Large labeling efforts or full-time QA programs.

Next Step

Review one workflow's current gates and monitoring, then define the first month's reliability plan. If you're pre-prod, start with an Audit or Build & Harden sprint.

Book an intro call:

calendly.com/philipstevens4/intro

Email: philipstevens4@gmail.com
Web: philipstevens.github.io

Scan to book a call

Not ready? Get the free Production Readiness Checklist to self-assess first.