// MODULE · OPERATIONS

The desk is the system.

Ten runbook procedures, six incident classes, six recovery trees, twelve monitors, and the twenty gates the cadence rides on.

10	06	12	20
RUNBOOKS	INCIDENTS	MONITORS	GATES

NEXURAL · SAGE IDEAS LLC
PUBLISHED 2026 · PERPLEXITY COMPUTER
nexural.com/learn/automation/operations

// table of contents

What you will learn.

01	Runbook · ten procedures across five categories	p. 03
02	Incidents · six classes & the four-step playbook	p. 09
03	Recovery trees · six guided decision flows	p. 13
04	Monitoring · twelve monitors across four layers	p. 20
05	Operating cadence · twenty gates, five cadences	p. 25
06	Honest operations principles	p. 31
S	Sources & further reading	p. 32

Each section is self-contained. If you only have time for one, read section 03 — the recovery trees are what you will reach for at 3 a.m. when something is wrong.

// section 01

Ten runbook procedures.

A runbook is what stops you from improvising in production. The ten procedures below cover deploy, rollback, incident response, drift, and broker operations — the categories every desk repeats. Each carries a trigger, the exact steps, a verify check, and a honest cadence.

Honest principle: if a procedure is not written down, it does not exist. The worst time to write a runbook is during an outage. The right time is the calm session after the last one.

ID	Category	Procedure
rolling-deploy	DEPLOY	Rolling deploy of new model version
feature-deploy	DEPLOY	Deploy a new feature into production
model-rollback	ROLLBACK	Rollback to previous model version
config-rollback	ROLLBACK	Rollback a config change
killswitch	INCIDENT	Pull the kill-switch
reconciliation	INCIDENT	Position reconciliation after partial outage
drift-investigation	DRIFT	Investigate a feature-drift alert
retrain	DRIFT	Scheduled model retrain
broker-failover	BROKER	Failover to backup broker
api-credentials	BROKER	Rotate broker API credentials

// section 01 · procedures 1-2 of 10

Procedures (1–2)

Rolling deploy of new model version

Category. DEPLOY · ID. `rolling-deploy` · Frequency. 1-2 times per month, per active model

Trigger

New model passed all 15 production gates and was approved by the model committee.

Steps

01 Tag the model and feature set: model_v1.4.0, features_v2.3.0

02 Deploy to shadow mode behind the live model for 3 trading days

03 Compare PnL, Sharpe, and turnover against the live model

04 Promote to 25% capital allocation; hold for 5 trading days

05 If no band exit at 25%, promote to 100% the next session

Verify

Live model_hash matches the tagged release in the model registry; shadow log shows ≤ 1.2x prediction divergence vs. the previous version.

Deploy a new feature into production

Category. DEPLOY · ID. `feature-deploy` · Frequency. Weekly cadence during research; monthly to production

Trigger

Feature passed offline validation: stable PSI, no look-ahead, contributes ≥ 5% to feature importance.

Steps

01 Backfill the feature for the full training window

02 Re-run cross-validation with the new feature added

03 If CV Sharpe improves, retrain the production model

04 Run 1 trading week in shadow before going live

05 Update the feature schema doc with the new filed_at timestamp

Verify

Feature is present in /api/predict requests for 100% of symbols; PSI < 0.1 between offline and live distributions in the first session.

// section 01 · procedures 3-4 of 10

Procedures (3–4)

Rollback to previous model version

Category. ROLLBACK · ID. `model-rollback` · Frequency. Rare — but practice quarterly on a paper-trading sandbox

Trigger

Live PnL exits the 95% backtest band for 3 consecutive days OR a P1 incident is filed against the current version.

Steps

01 Flip the model_version flag in production config back to the prior tag

02 Confirm the prior model_hash is still pinned and reachable in the registry

03 Restart the inference workers; check the first 100 predictions for sanity

04 Notify the on-call channel with the reason and reference to the incident ticket

Verify

Production /healthz reports the prior model_hash; PnL on the first session post-rollback returns to within the backtest band.

Rollback a config change

Category. ROLLBACK · ID. `config-rollback` · Frequency. As needed; aim for < 1 per quarter

Trigger

Risk service or PnL monitor alerts within 60 minutes of a config change being applied.

Steps

01 git revert the offending commit; force-push is forbidden

02 Redeploy with the revert; confirm the live config hash matches the revert commit

03 Inspect any positions opened in the affected window for exposure mismatches

04 File a hot-fix ticket with the original change author for a proper redo

Verify

git rev-parse HEAD on the live host matches the revert commit; no further alerts in the next 30 minutes.

// section 01 · procedures 5-6 of 10

Procedures (5–6)

Pull the kill-switch

Category. INCIDENT · ID. `killswitch` · Frequency. 0 to a few times per year. Drill quarterly.

Trigger

Any P0 incident: broker outage, risk breach, stale data, runaway model behavior.

Steps

01 Hit the kill-switch endpoint or physical button — both are equivalent

02 Cancel all open orders across all venues; do NOT wait for venue confirmation

03 Snapshot the current position book, the model_hash, and the last 10 minutes of predictions

04 Open an incident channel; assign an incident commander before triaging the cause

Verify

Order log shows zero new orders for ≥ 60 seconds after the switch; venue-side open-order count is zero.

Position reconciliation after partial outage

Category. INCIDENT · ID. `reconciliation` · Frequency. After every P0 incident; spot-check monthly

Trigger

Order acknowledgements went silent for any window during a trading session.

Steps

01 Pull the canonical position file from the broker, not your own order log

02 Diff against the internal position state at the start of the silence window

03 Manually classify any discrepancy: filled but not reported, reported but not filled, duplicate

04 Adjust internal state to match broker truth; do not let the model recover from a wrong position

Verify

Broker position file and internal state agree to the share; the next session opens from a single source of truth.

// section 01 · procedures 7-8 of 10

Procedures (7–8)

Investigate a feature-drift alert

Category. DRIFT · ID. `drift-investigation` · Frequency. Weekly review of all production features

Trigger

PSI > 0.25 on any production feature, OR Sharpe contribution of a top-3 feature halves over a rolling 30-day window.

Steps

01 Plot the live and training distributions side-by-side for the offending feature

02 Check if the cause is pipeline (new vendor format, missing column) or genuine regime

03 Pipeline: roll back. Regime: schedule a controlled retrain with the new data included

04 If neither, audit the feature for hidden look-ahead — drift is often the late discovery of leakage

Verify

PSI returns under 0.15 within one trading week of the fix, OR the model is officially retired.

Scheduled model retrain

Category. DRIFT · ID. `retrain` · Frequency. Quarterly minimum

Trigger

Quarterly cadence OR drift event resolved with 'retrain' decision.

Steps

01 Refresh the training window by N most-recent days (defined per model)

02 Re-run purged k-fold CV; require deflated Sharpe ≥ the existing model

03 Deploy via the rolling-deploy procedure, not a hot replace

04 Archive the prior model_hash with its full training data hash and date stamps

Verify

New model out-of-sample Sharpe within 10% of the prior model; predictions agree on ≥ 70% of symbols on day one.

// section 01 · procedures 9-10 of 10

Procedures (9–10)

Failover to backup broker

Category. BROKER · ID. `broker-failover` · Frequency. Test on a paper account monthly

Trigger

Primary broker heartbeat fails for 3 consecutive checks, OR scheduled maintenance window.

Steps

01 Flip the broker_route flag to the secondary venue

02 Confirm authentication on the secondary; check rate-limit headroom

03 Open a single-share test order on a liquid symbol; cancel immediately

04 Resume trading; reconcile any pending orders from the primary

Verify

First five orders on the secondary fill within the expected latency band; primary's status page confirms outage.

Rotate broker API credentials

Category. BROKER · ID. `api-credentials` · Frequency. Quarterly

Trigger

Quarterly cadence, OR any credential exposure event (commit to a public repo, lost laptop).

Steps

01 Generate the new credential in the broker UI; do not delete the old one yet

02 Push the new credential to the secret manager with a future activation timestamp

03 Verify the new credential works by issuing a non-mutating call (account balance)

04 Activate; wait 5 minutes; revoke the old credential

Verify

Live trading uses the new credential hash; broker UI shows the old credential as revoked.

// section 02

Six incident classes.

Six categories cover ~95% of real incidents on an automated trading desk. Severity decides whether you can throttle or must stop. The playbook decides what comes next.

ID	Class	Severity	One-liner
broker-outage	Broker / venue outage	P0	Order acknowledgements stop. Positions of unknown state.
feature-drift	Feature drift detected	P1	Live feature distribution diverges from training. PSI > 0.25.
pnl-divergence	Live PnL exits backtest band	P1	Daily PnL more than 2σ below backtest expectation for 3+ days.
stale-data	Stale or missing data feed	P0	Bar timestamp older than 2x bar interval. Decisions on yesterday's price.
config-drift	Config drift / unintended change	P1	Live config diverges from the committed YAML in version control.
risk-breach	Risk-limit breach	P0	Position, drawdown, or VaR limit crossed.

P0 means stop trading. P1 means degrade — typically throttle to 25% size until the picture clears. P2 means monitor; act on the next planned cycle. The severity is non-negotiable; the response is the playbook.

// section 02 · incidents 1-2 of 6

Incident playbook (1–2)

Broker / venue outage

ID. `broker-outage` · Severity. P0 · One-liner. Order acknowledgements stop. Positions of unknown state.

Detection

Heartbeat to the broker REST/FIX endpoint fails 3 times in 30 seconds, OR order-acknowledgement latency exceeds 5x baseline.

Immediate action

Pull the kill-switch. Stop generating new orders. Do NOT retry blindly — duplicate fills are the single most expensive recovery mistake.

Containment

Switch to the broker's status page and any backup venue. Reconcile open positions against the last known good order log. Treat anything ambiguous as 'unknown until manually confirmed'.

Postmortem

Document the gap window (first failed heartbeat to first successful reconciliation). Add the broker outage to your annual SLA review. Decide if a second venue is now mandatory.

Feature drift detected

ID. `feature-drift` · Severity. P1 · One-liner. Live feature distribution diverges from training. PSI > 0.25.

Detection

Population Stability Index between training distribution and rolling 5-day live distribution exceeds 0.25 on any top-10 feature.

Immediate action

Throttle live size to 25% until the cause is known. Disable any model whose top-3 features show PSI > 0.25.

Containment

Determine if the cause is a data-pipeline change (new vendor format, missing field) or genuine regime shift. If pipeline: roll back. If regime: schedule a controlled retrain.

Postmortem

Add the drift event to the model registry. If this is the 2nd drift on the same model in 90 days, the model retires on schedule rather than relaxing the threshold.

// section 02 · incidents 3-4 of 6

Incident playbook (3–4)

Live PnL exits backtest band

ID. `pnl-divergence` · Severity. P1 · One-liner. Daily PnL more than 2σ below backtest expectation for 3+ days.

Detection

Live cumulative PnL drops below the 5th-percentile band of the backtest distribution for 3 consecutive trading days.

Immediate action

Halve live size. Log the divergence with the model_hash + feature_hash + lib_hash so it can be reproduced.

Containment

Run the same period through the backtest engine with live fills. If backtest agrees: signal is genuinely decaying. If backtest disagrees: a production bug.

Postmortem

Three outcomes: retire, retrain, or refit. Document which and why. No model survives two consecutive band exits without a real change.

Stale or missing data feed

ID. `stale-data` · Severity. P0 · One-liner. Bar timestamp older than 2x bar interval. Decisions on yesterday's price.

Detection

Latest market-data bar is older than 2x the configured bar interval (e.g., > 2 min on a 1-min feed) for any traded symbol.

Immediate action

Block new orders for the affected universe. Open positions: tighten stops and let existing flat by EOD if data does not recover.

Containment

Failover to the backup data vendor. Confirm the failure scope: one symbol, one venue, or full feed. Reconcile any orders sent in the affected window.

Postmortem

Add the symbol-level age check to the pre-trade gate if it was not already there. Confirm the backup feed actually carried the missing window.

// section 02 · incidents 5-6 of 6

Incident playbook (5–6)

Config drift / unintended change

ID. `config-drift` · Severity. P1 · One-liner. Live config diverges from the committed YAML in version control.

Detection

Scheduled job hashes /etc/strategy/config.yml against the deployed git ref. Mismatch triggers an alert within 5 minutes.

Immediate action

Pause any model whose config is mismatched. Snapshot the current live config and the expected config side-by-side.

Containment

Diff the two configs. If the live one was a hot-fix nobody committed, commit it now. If the live one is wrong, redeploy from the committed ref.

Postmortem

Disable manual edits to production config files. All config changes flow through a PR + redeploy. Add the offending field to the alerting allowlist if it's expected to be live-tuned.

Risk-limit breach

ID. `risk-breach` · Severity. P0 · One-liner. Position, drawdown, or VaR limit crossed.

Detection

Real-time risk service flags a hard-limit breach: max position size, daily drawdown floor, 1-day VaR ceiling, or single-name concentration.

Immediate action

Trigger the kill-switch. The risk service auto-flattens to the next compliant size; do not let traders override it.

Containment

Determine the cause: sizing bug, signal explosion, or market gap. If sizing bug: stay flat until the patch lands. If signal: review the next 4 hours of intended trades before resuming.

Postmortem

Every risk breach gets a written incident review within 24 hours. The threshold that was breached does not get raised as part of the fix.

// section 03

Six recovery trees.

Decision trees for "something is wrong, what now?". Pick the symptom; answer the questions; the tree walks down to a concrete action. Five leaf kinds: kill-switch, throttle, rollback, investigate, resume. The tree never lets you skip reconciliation, never relaxes a risk limit silently, and never resumes on hope.

The point of these trees is not to be exhaustive — it is to be repeatable. The same symptom should produce the same action regardless of who is on call and what time of night it is.

// recovery tree · no-fills

Orders are not filling

One-liner. Order route looks healthy but acknowledgements are silent.

Decision tree

```
Q1: Is the broker status page green?
  NO  --> [KILL-SWITCH]  Broker is down. Do not retry blindly —
                         duplicate fills on resume are the most
                         expensive mistake here.
  YES --> Q2: Are orders respecting tick / lot size?
            NO  --> [ROLLBACK]  Order-builder bug. Revert the
                                last change touching order
                                construction. Re-test with one share.
            YES --> Q3: Has the rate-limit budget been exhausted?
                      NO  --> [INVESTIGATE]  Check venue reject codes.
                                             Open a broker ticket if
                                             codes are blank.
                      YES --> [THROTTLE]  Halve order frequency. Wait
                                          one rate-limit window. Resume
                                          at 25% size to verify.
```

// recovery tree · wrong-pnl

Live PnL diverges from backtest

One-liner. Daily PnL outside the 95% backtest band for multiple days.

Decision tree

```
Q1: Has this gone on for 3 or more trading days?
  NO  --> [INVESTIGATE]  One or two days outside the band is
                         expected ~5% of the time. Log it; continue
                         monitoring; do not over-react.
  YES --> Q2: When you replay the same window through the
              backtest engine with live fills, do they agree?
            YES --> [THROTTLE]  Signal is genuinely decaying.
                                Halve allocation. Schedule a retire /
                                retrain decision within the week.
            NO  --> [ROLLBACK]  Production bug — engine and live
                                disagree. Rollback to the prior
                                model_hash and file a bug ticket.
```

// recovery tree · missing-data

Market-data feed is stale or missing

One-liner. Bar timestamps older than 2x the bar interval.

Decision tree

```
Q1: Is the staleness on a single symbol?
  YES --> Q2: Did the symbol have a corporate action today
              (split, dividend, halt)?
            YES --> [INVESTIGATE]  Halt or corporate-action gap.
                                   Suppress the symbol; resume the
                                   rest of the universe.
            NO  --> [INVESTIGATE]  Vendor-side single-symbol issue.
                                   Drop the symbol; resume universe;
                                   file a vendor ticket.
  NO  --> Q3: Is the backup data vendor healthy?
            YES --> [ROLLBACK]   Failover to the backup vendor.
                                 Reconcile any orders sent during
                                 the stale window.
            NO  --> [KILL-SWITCH] Both vendors down. Stop trading.
                                  Tighten stops; flatten by EOD if
                                  data does not recover.
```

// recovery tree · alerts-spam

Alert channel is on fire

One-liner. Dozens of alerts in minutes; signal-to-noise collapsing.

Decision tree

```
Q1: Is there a single root cause behind &gt; 60% of the alerts?
  NO  --> [THROTTLE]  Multiple unrelated issues. Halve live size;
                      triage alerts in priority order; do not
                      ignore the noisy ones — that is how the next
                      outage starts.
  YES --> Q2: Is the root cause already a known incident with a
              ticket open?
            YES --> [INVESTIGATE]  Snooze downstream alerts that
                                   depend on the parent incident.
                                   Keep the parent alert live.
            NO  --> [INVESTIGATE]  Open a parent incident; link the
                                   duplicated alerts; assign an
                                   incident commander.
```

// recovery tree · config-mismatch

Live config does not match git

One-liner. Config-drift check failed.

Decision tree

```
Q1: Was a deploy or hot-fix performed in the last 2 hours?
  YES --> Q2: Was the change intentional?
            YES --> [INVESTIGATE]  Commit the live config now. Push
                                   the commit. Re-run the drift
                                   check; it should pass.
            NO  --> [ROLLBACK]    Accidental hot-edit. Redeploy from
                                  the committed ref. Audit who has
                                  prod-write access.
  NO  --> [ROLLBACK]  Unexplained drift. Redeploy from the
                      committed ref. Open a security review —
                      unexpected prod writes are an audit-trail issue.
```

// recovery tree · post-incident

Coming back from a stop

One-liner. Incident closed; ready to resume.

Decision tree

```
Q1: Is the root cause documented and the fix verified in a
    non-prod environment?
  NO  --> [INVESTIGATE]  Do not resume on hope. A documented root
                         cause plus a verified fix in staging are
                         the minimum.
  YES --> Q2: Have positions been reconciled to broker truth?
            NO  --> [INVESTIGATE]  Reconcile to the broker file
                                   first. Trading from an unknown
                                   book is how small incidents
                                   become large ones.
            YES --> Q3: Is risk service operating with the original
                        (un-relaxed) limits?
                      NO  --> [INVESTIGATE]  Limits got softened
                                             during the incident.
                                             Restore them first.
                      YES --> [RESUME]  Resume at 25% size for one
                                        full session. Promote to
                                        100% only if incident-free.
```

// section 04

Twelve monitors. Four layers.

Monitoring is what keeps the system observable. Twelve concrete monitors across four layers — infrastructure, data, model, PnL — each with a metric, an alert threshold, a channel, and a frequency. Pageable alerts wake on-call. Slack alerts queue. The split decides whether you sleep.

Layer	Metric	Threshold	Channel	Freq.
INFRA	Inference worker liveness	Healthcheck fails 2x in 60s	PAGE	every 30s
INFRA	Broker heartbeat latency	p95 > 500ms for 3 windows	PAGE	every 10s
INFRA	Job queue depth	> 1000 unprocessed	SLACK	every 1m
DATA	Market-data bar age	> 2x bar interval, any symbol	PAGE	every 5s
DATA	Feature PSI vs. training	PSI > 0.25 on any top-10 feature	SLACK	daily
DATA	Vendor-feed reconciliation	Bar count mismatch primary vs. backup	SLACK	hourly
MODEL	Prediction distribution	KS-stat vs. baseline > 0.15	SLACK	hourly
MODEL	Feature-importance churn	Top-5 set changes by ≥ 2 features	SLACK	weekly
MODEL	Inference latency	p99 > 50% of decay horizon	PAGE	every 1m
PNL	Daily PnL vs. backtest band	Outside 95% band 3 days in a row	PAGE	daily
PNL	Slippage vs. expected	Realised - expected > 2 bps for 5 days	SLACK	daily
PNL	Per-symbol concentration	> 25% of daily PnL from one name	SLACK	daily

// section 04 · layer · infra

Infrastructure

Inference worker liveness

Threshold. `Healthcheck fails 2x in 60s`
Channel. PAGE · Frequency. `every 30s`

/healthz endpoint must respond 200 with the live model_hash. Two consecutive failures pages on-call immediately.

Broker heartbeat latency

Threshold. `p95 > 500ms for 3 windows`
Channel. PAGE · Frequency. `every 10s`

REST or FIX ping latency. Three consecutive 1-minute windows over threshold triggers failover evaluation.

Job queue depth

Threshold. `> 1000 unprocessed`
Channel. SLACK · Frequency. `every 1m`

Backfills, retrains, and shadow runs queue up here. Sustained depth means a worker is wedged.

// section 04 · layer · data

Data

Market-data bar age

Threshold. `> 2x bar interval, any symbol`
Channel. PAGE · Frequency. `every 5s`

Stale data risks decisions on yesterday's price. Symbol-level so a single halt does not silence the whole alert.

Feature PSI vs. training

Threshold. `PSI > 0.25 on any top-10 feature`
Channel. SLACK · Frequency. `daily`

Population Stability Index between live and training feature distributions. Anything over 0.25 demands an investigation.

Vendor-feed reconciliation

Threshold. `Bar count mismatch primary vs. backup`
Channel. SLACK · Frequency. `hourly`

Count of bars received from primary and secondary vendor for the same symbol/window must agree. Disagreements caught early prevent failover surprises.

// section 04 · layer · model

Model

Prediction distribution

Threshold. `KS-stat vs. baseline > 0.15`
Channel. SLACK · Frequency. `hourly`

Two-sample Kolmogorov-Smirnov test between today's predictions and a rolling 30-day baseline. Drift here is the strongest early signal of edge decay.

Feature-importance churn

Threshold. `Top-5 set changes by ≥ 2 features`
Channel. SLACK · Frequency. `weekly`

Recompute feature importance over the most recent 30 days. Sudden churn in the top-5 set demands a retrain or retire decision.

Inference latency

Threshold. `p99 > 50% of decay horizon`
Channel. PAGE · Frequency. `every 1m`

A signal decaying in 5 minutes must reach the broker in well under 2.5 minutes including inference. p99 is the right percentile here, not the median.

// section 04 · layer · pnl

PnL

Daily PnL vs. backtest band

Threshold. `Outside 95% band 3 days in a row`
Channel. PAGE · Frequency. `daily`

Cumulative live PnL plotted on the backtest distribution. A 3-day band exit is the textbook trigger for a retire/retrain conversation.

Slippage vs. expected

Threshold. `Realised - expected > 2 bps for 5 days`
Channel. SLACK · Frequency. `daily`

Compare realised fill price against the price assumed in the backtest cost model. Persistent slippage is real money and an early sign the cost model is stale.

Per-symbol concentration

Threshold. `> 25% of daily PnL from one name`
Channel. SLACK · Frequency. `daily`

Even a profitable day with single-name dominance is a risk warning. Either size down that name or document why the concentration is intentional.

// section 05

Operating cadence.

Twenty gates across five cadences. The pre-launch list runs once before any capital moves. The daily list runs every session. Weekly, monthly, and quarterly cycles each carry their own gates. Audit cadence-by-cadence — not item-by-item.

ID	Cadence	Gate
P1	PRE-LAUNCH	Kill-switch tested on a paper run within the last 7 days
P2	PRE-LAUNCH	Risk limits committed in git; live config hash matches
P3	PRE-LAUNCH	Broker API credentials rotated and tested
P4	PRE-LAUNCH	Backup data vendor authenticated and current
D1	DAILY	Pre-market data freshness check
D2	DAILY	Model_hash matches the registry's pinned production version
D3	DAILY	PnL band check against backtest distribution
D4	DAILY	End-of-day position reconciliation to broker truth
D5	DAILY	Slippage and fill-quality report reviewed
W1	WEEKLY	Feature drift report (PSI on all top-10 features)
W2	WEEKLY	Alert-noise audit
W3	WEEKLY	Vendor heartbeat / SLA review
W4	WEEKLY	Incident-ticket grooming
M1	MONTHLY	Disaster-recovery drill on a non-prod sandbox
M2	MONTHLY	Backup integrity test (restore-from-cold check)
M3	MONTHLY	Cost-model recalibration
M4	MONTHLY	Capacity and headroom review
Q1	QUARTERLY	Per-model retire / retrain / refit decision
Q2	QUARTERLY	Risk-limit framework review with stakeholders
Q3	QUARTERLY	Access-and-audit review

// section 05 · cadence · pre

Pre-launch

P1 · Kill-switch tested on a paper run within the last 7 days

Physical or endpoint kill-switch verified to flatten all positions and block new orders within 60 seconds. Drill report filed.

P2 · Risk limits committed in git; live config hash matches

Max position size, daily drawdown floor, daily VaR ceiling, and single-name concentration cap all defined in /etc/risk/limits.yml and matching the deployed hash.

P3 · Broker API credentials rotated and tested

Rotation in the last 90 days. New credentials verified with a non-mutating call before the old ones were revoked.

P4 · Backup data vendor authenticated and current

Secondary feed has run in parallel for at least 5 trading days with < 0.5% bar-count divergence vs. the primary.

// section 05 · cadence · daily

Daily

D1 · Pre-market data freshness check

All traded symbols have a bar within the last 2x interval as of the open. Any stale symbol is suppressed before orders begin.

D2 · Model_hash matches the registry's pinned production version

Live /healthz reports the same model_hash as the production tag in the model registry. Mismatch blocks trading.

D3 · PnL band check against backtest distribution

Cumulative live PnL plotted on the backtest distribution. Document whether today landed inside the 95% band.

D4 · End-of-day position reconciliation to broker truth

Broker position file matches internal state to the share. Reconciliation report archived for audit.

D5 · Slippage and fill-quality report reviewed

Realised vs. expected fill price; rejected order count; average fill latency. Document any deviation > 1σ.

// section 05 · cadence · weekly

Weekly

W1 · Feature drift report (PSI on all top-10 features)

Population Stability Index between live and training feature distributions. Anything ≥ 0.25 opens a drift ticket.

W2 · Alert-noise audit

Count of alerts paged + Slack-sent. Threshold reviews for any alert firing > 5 times without an action.

W3 · Vendor heartbeat / SLA review

Both market-data and broker SLAs met for the week. File any breach with the vendor before it lapses.

W4 · Incident-ticket grooming

All P1/P2 tickets from the prior week have an assigned owner and a written postmortem if closed.

// section 05 · cadence · monthly

Monthly

M1 · Disaster-recovery drill on a non-prod sandbox

Practice a full broker outage + failover + reconciliation cycle on paper trading. Time-to-recovery target documented.

M2 · Backup integrity test (restore-from-cold check)

Pull a random model_hash and a random feature snapshot from cold storage; verify both load and predict on the test set.

M3 · Cost-model recalibration

Compare realised slippage and fees against the cost model used in the backtest. If the gap exceeds 1 bp, retune.

M4 · Capacity and headroom review

ADV consumed per symbol, rate-limit budget used, infra resource headroom. Document any line approaching 70%.

// section 05 · cadence · quarterly

Quarterly

Q1 · Per-model retire / retrain / refit decision

Every active model labelled with one of the three. No model lives indefinitely by default.

Q2 · Risk-limit framework review with stakeholders

Max position, drawdown, VaR, concentration. Limits documented; any raise requires a written justification.

Q3 · Access-and-audit review

Who has prod-write access; who has approval rights for deploys; quarterly attestation signed.

// section 06

Honest operations principles.

If you take only ten ideas from this PDF, take these.

01	If a procedure is not written down, it does not exist. Document yours; do not improvise live.
02	Severity is non-negotiable. P0 stops trading; P1 throttles; P2 monitors. Do not negotiate with the on-call playbook.
03	When in doubt, kill the switch. Duplicate fills on a botched recovery are the most expensive mistake in this business.
04	Reconcile to broker truth, never to your own log. Your log is what you intended; the broker file is what happened.
05	Pageable alerts wake people; Slack alerts queue. Choose carefully. An alert that pages nobody is decoration.
06	Drift is often the late discovery of leakage. If a feature drifts hard, audit it for look-ahead before retraining.
07	No model lives forever by default. Quarterly retire/retrain/refit reviews are mandatory, not optional.
08	Risk limits do not get raised as part of an incident fix. The threshold that was breached stays where it was.
09	Backups are restore-tested or they do not exist. Run a cold-restore drill every month, not just when you need it.
10	Postmortems are blameless and written. Every incident produces an artifact; the artifact is more valuable than the fix.

Ten lines. If they feel obvious, they should — every one of them has been violated by someone, expensively, in the last decade. The point of writing them down is to make the obvious enforceable.

// further reading

Sources and further reading.

[1] Google SRE Book — chapters on incident response and postmortems.

https://sre.google/sre-book/table-of-contents/

[2] Google SRE Workbook — practical runbook and on-call practices.

https://sre.google/workbook/table-of-contents/

[3] Etsy — Blameless PostMortems and a Just Culture.

https://www.etsy.com/codeascraft/blameless-postmortems

[4] PagerDuty — Incident Response Documentation.

https://response.pagerduty.com/

[5] ITIL Foundation — Change Management Practice.

https://www.axelos.com/certifications/propath/itil-4-foundation

[6] Twelve-Factor App — config & backing services.

https://12factor.net/

[7] AWS Well-Architected — Reliability Pillar.

https://docs.aws.amazon.com/wellarchitected/latest/reliability-pillar/welcome.html

[8] Google — Postmortem Culture: Learning from Failure.

https://sre.google/sre-book/postmortem-culture/

[9] CFTC Regulation Automated Trading (Reg AT) — risk controls.

https://www.cftc.gov/sites/default/files/idc/groups/public/@lrfederalregister/documents/file/2015-29842a.pdf

[10] SEC Rule 15c3-5 — Market Access Rule.

https://www.sec.gov/rules/final/2010/34-63241.pdf

[11] López de Prado — Advances in Financial Machine Learning (production chapters).

https://www.wiley.com/en-us/Advances+in+Financial+Machine+Learning-p-9781119482086

[12] PSI / Population Stability Index — drift detection primer.

https://www.listendata.com/2015/05/population-stability-index.html

Continue reading → Next module: PDF Library