

Reality Anchor — Probe Daemons and Watchdog

Reality Anchor — Probe Daemons and Watchdog

Status: DRAFT (MC #101450, 2026-05-19) **Author:** FlowForge / Kelsey Hightower **Doctrine:** Reality Anchor v1 (approved 2026-05-15, docs.alai.no/books/system-architecture/page/reality-anchor-doctrine-v1-final)

Why This Exists

The Reality Anchor doctrine (2026-05-15) established that probe output IS evidence — deterministic tool output, not LLM inference. Two probe daemons were deployed to provide continuous fleet health signals:

- `com.john.auto-verify-regression` — regression suite against the anti-hallucination probe library
- `com.john.ollama-health-probe` — Ollama fleet health (ANVIL + FORGE endpoints)

In the week of 2026-05-11 to 2026-05-18, both daemons stopped producing fresh state output. Root cause: `auto-verify-regression` was scheduled at `StartCalendarInterval` (once daily at 06:00) rather than a continuous interval. Combined with the absence of a watchdog, there was no circuit-breaker to detect and recover from the audit blind spot.

This document describes the fix applied under MC #101450 and the ongoing watchdog architecture.

Daemon Inventory

1. `com.john.auto-verify-regression`

Property	Value
Plist	~/Library/LaunchAgents/com.john.auto-verify-regression.plist
Script	~/system/tools/auto-verify-regression.js
Interval	900 seconds (15 minutes) — changed from daily StartCalendarInterval
RunAtLoad	true
Stdout log	~/system/logs/auto-verify-regression.log
State written	~/system/logs/auto-verify-regression.log (tail -1 = regression result)

What it does: Runs the 5-probe regression suite against the anti-hallucination probe library. Each probe runs a known-bad case (expected FAIL) and a known-good case (expected PASS). Emits `5/5 PASS` or lists failures. Failure = evidence pipeline degraded.

2. com.john.ollama-health-probe

Property	Value
Plist	~/Library/LaunchAgents/com.john.ollama-health-probe.plist
Script	~/system/tools/ollama-health-probe.sh
Interval	60 seconds (unchanged)
RunAtLoad	true
Stdout log	~/system/logs/ollama-health-probe.out
State written	~/system/state/ollama-fleet.json

What it does: Probes `localhost:11434` (ANVIL) and `10.0.0.2:11434` (FORGE) via `GET /api/tags`. Writes JSON status (healthy/degraded/down) to `ollama-fleet.json`. Sends Slack alert to `#ops` on status transitions. DEGRADED = primary down, backup (Tailscale) up.

3. com.john.reality-anchor-watchdog (NEW — MC #101450)

Property	Value
Plist	~/Library/LaunchAgents/com.john.reality-anchor-watchdog.plist
Script	~/system/tools/reality-anchor-watchdog.sh
Interval	3600 seconds (1 hour)

Property	Value
RunAtLoad	true
Alert log	~/.cache/reality-anchor-stale-alerts.log

What it does: Checks mtime of each probe's state file every hour. If any state file has not been written in > 24 hours, it:

1. Logs `STALE_PROBE_ALERT` to `~/.cache/reality-anchor-stale-alerts.log`
2. Calls `launchctl start <daemon>` for one auto-restart attempt
3. Logs the restart result (success or escalation-needed)

If state is fresh, logs `OK` with current age.

Alert Path

```
Probe state file mtime > 24h
  → reality-anchor-watchdog fires
    → ~/.cache/reality-anchor-stale-alerts.log (STALE_PROBE_ALERT line)
    → launchctl start <probe> (auto-restart attempt)
    → if restart fails: "ESCALATION NEEDED" logged
```

Manual escalation path:

```
grep "ESCALATION NEEDED" ~/.cache/reality-anchor-stale-alerts.log
  → Slack #ops manual alert
  → CEO notification if probe offline > 48h
```

Future: connect `reality-anchor-stale-alerts.log` growth to a Slack webhook. When file size increases since last check cycle, post to `#ops`. This closes the loop from watchdog to human-visible alert without requiring a separate daemon.

Recovery Runbook

If probes are stale:

```
# 1. Check state
launchctl list | grep -E "auto-verify-regression|ollama-health-probe|reality-anchor-watchdog"
cat ~/.cache/reality-anchor-stale-alerts.log | tail -20
```

```
# 2. Manual restart (watchdog does this automatically, but for immediate action)
launchctl start com.john.auto-verify-regression
launchctl start com.john.ollama-health-probe

# 3. Verify within 60s
ls -lat ~/system/state/ollama-fleet.json ~/system/logs/auto-verify-regression.log

# 4. If plist is unloaded (not listed at all):
launchctl load ~/Library/LaunchAgents/com.john.auto-verify-regression.plist
launchctl load ~/Library/LaunchAgents/com.john.ollama-health-probe.plist
launchctl load ~/Library/LaunchAgents/com.john.reality-anchor-watchdog.plist
```

E2E Test

Proveo validation test: `~/system/tests/reality-anchor-recovery-test.sh`

- `--dry-run` flag: mocks destructive steps (safe for CI / scheduled validation)
- Live mode: requires operator confirmation before stopping Ollama
- Tests A (stop detection), B (recovery detection), C (watchdog stale alert)

Run: `bash ~/system/tests/reality-anchor-recovery-test.sh --dry-run`

Change Log

Date	Change	MC
2026-05-15	Reality Anchor doctrine approved; probes deployed	#100818-#100833
2026-05-19	auto-verify-regression interval changed to 900s; watchdog created	#101450

Revision #2

Created 2026-05-19 15:55:14 UTC by John

Updated 2026-06-21 20:03:48 UTC by John