

pi-orchestrator: john H-task auto-pause root cause + #10063 reconcile

pi-orchestrator: john H-task auto-pause root cause + #10063 reconcile

Created: 2026-05-02

MC References: [#10063](#) (phantom fix), [#10517](#) (true fix)

Daemon: `com.john.pi-orchestrator` (currently STOPPED, reactivation pending CEO Step 3)

Symptom

John's H-priority tasks were being auto-paused without user action. The pi-orchestrator daemon would intercept high-priority john tasks and route them through `queueForHuman` instead of executing them, creating a silent work-stoppage pattern.

Investigation Finding — Phantom Fix in MC #10063

MC #10063 (2026-04-XX) claimed to fix the auto-pause behavior by adding configuration flags:

- `skip_interactive_owners: ["john", "alem"]`
- `interactive_grace_seconds: 300`

Problem: These config keys were specified in the task's acceptance criteria and marked COMPLETE, but were **never actually written to** `~/system/config/pi-orchestrator-config.json`.

Anti-pattern identified: "Proveo PASS but code doesn't match documentation" — the validation passed based on spec intent rather than verifying actual configuration state.

True Root Cause

The mechanism **actually** auto-pausing john H-tasks was a **dead fallback block** in `~/system/kernel/pi-orchestrator.js`:

```
// Original lines 3409-3421 (13 lines, now removed)
if (!selectedTask) {
  // Fallback: check for john tasks
  const johnTask = execSync(
    'node ~/system/tools/mc.js next-task --owner john',
    { encoding: 'utf8' }
  ).trim();

  if (johnTask) {
    queueForHuman(johnTask);
    return null;
  }
}
```

When task selection failed (empty queue or filter mismatch), this fallback would:

1. Synchronously fetch the next john task via `mc.js next-task --owner john`
2. Queue it for human review via `queueForHuman()`
3. Return `null`, preventing execution

This created the observed auto-pause behavior **regardless of the missing config flags**.

Fix Applied — MC #10517

Date: 2026-05-02

Builder: Codecraft

Validator: Proveo

Changes:

1. **Configuration reconciliation** — Added missing flags to `~/system/config/pi-orchestrator-config.json` at lines 93-94:

```
"skip_interactive_owners": ["john", "alem"],  
"interactive_grace_seconds": 300
```

2. **Dead fallback removal** — Replaced the 13-line `execSync` fallback block in `~/system/kernel/pi-orchestrator.js` (original lines 3409-3421) with a 4-line comment + null return:

```
// No fallback to john tasks – auto-pause removed per MC #10517.  
// Configuration now controls interactive routing via skip_interactive_owners.  
log('No task selected; returning null.');
```

```
return null;
```

Verification

Proveo validation: APPROVED 2026-05-02

Acceptance Criteria: 4/4 PASS

- **AC1:** `pi-orchestrator-config.json` contains `skip_interactive_owners: ["john", "alem"]`
- **AC2:** `pi-orchestrator-config.json` contains `interactive_grace_seconds: 300`
- **AC3:** Dead fallback block removed from `pi-orchestrator.js` (lines 3409-3421 replaced)
- **AC4:** No `execSync` call to `mc.js next-task --owner john` in the selection path

Evidence:

- Config diff: `git diff ~/system/config/pi-orchestrator-config.json`
- Code diff: `git diff ~/system/kernel/pi-orchestrator.js`
- No remaining `queueForHuman` calls in fallback path: `grep -n "queueForHuman" ~/system/kernel/pi-orchestrator.js` shows only intentional usage in interactive routing logic

Daemon State

Current state: `com.john.pi-orchestrator` is **STOPPED** (unloaded via `launchctl unload`).

Reactivation: Pending CEO Step 3 directive. **DO NOT restart daemon** until explicitly approved — this is part of a phased rollout to validate the fix does not introduce regression.

To check status:

```
launchctl list | grep pi-orchestrator  
# Empty output = daemon not loaded
```

To restart (when authorized):

```
launchctl load ~/Library/LaunchAgents/com.john.pi-orchestrator.plist  
tail -f ~/system/logs/pi-orchestrator.log
```

Cross-References

- **MC #10063:** Original task claiming fix (phantom — config never written, Proveo validated spec not state)
- **MC #10517:** True fix reconciling config + removing dead fallback (Proveo APPROVED 2026-05-02)
- **Related pattern:** Feedback memo `feedback_task_description_state_verify.md` — agents must tool-verify state before writing it into MC descriptions or acceptance criteria

Lessons

1. **Proveo must verify actual state, not spec intent.** A config flag in the task description \neq the flag exists in the file.
2. **Dead code can be the true mechanism.** The "fix" in #10063 was irrelevant because the real culprit was a fallback block that ran regardless of config.
3. **Daemon restart \neq verification.** Stopping the daemon masked the symptom but didn't prove the fix. Reactivation under observation is the true test.

Generated by Skillforge for MC #10517 documentation deliverable. HiveMind sync pending.

Revision #2

Created 2026-05-02 12:29:40 UTC by John

Updated 2026-06-07 20:00:56 UTC by John