

Multi-Session Isolation — T10-quad Validation

T10-quad Validation Report — Phase 3 P1 Isolation Sweep

****Owner:**** Proveo (Angie Jones)

****Date:**** 2026-05-18

****MC:**** #101336

****Input:**** phase3-p1-sweep-log.md (CodeCraft MC #101335)

****Top Verdict:**** PASS

Top-Level Verdict: PASS

All 6 P1 harness fixes verified SAFE. All 3 deprecation warn hooks fire on legacy paths with backward compat preserved. Both append paths SAFE under 4-concurrent-writer load. MEMORY.md daemon serialises correctly (tmp clone test). All 3 DEFERRED items confirmed honestly tagged with grep-verified rationale.

Track 1 — New P1 Harness Fixes (P1-1, P1-8, P1-9)

```
**Command:** `diagnose-  
session-collision.sh --targets  
lightrag_health_lockf,opus_token  
_cas,bash_token_cas --writers  
4`
```

```
**Results from probe.jsonl**  
(`/tmp/session-collision-  
20260518T201729/probe.jsonl`):
```

```
| Target | Verdict | Expected |  
Match |  
|-----|-----|-----|-----|  
| lightrag_health_lockf | SAFE |  
fire_count_total == pre+4 | YES |  
| opus_token_cas | SAFE |  
exactly 1 winner of mv race |
```

YES |
| bash_token_cas | SAFE |
exactly 1 winner of mv race |
YES |

****Probe.jsonl line citations
(extracted verdicts):****

- `lightrag_health_lockf: SAFE`
- `opus_token_cas: SAFE`
- `bash_token_cas: SAFE`

****Track 1 Verdict: PASS****

**## Track 2 — Legacy
Regression (P1-1 contrast)**

```
**Command:** `diagnose-session-collision.sh --targets lightrag_health --writers 4`
```

```
**Result from probe.jsonl**  
(`/tmp/session-collision-20260518T201737/probe.jsonl`):  
- `lightrag_health:  
LAST_WRITER_WINS`
```

Confirms: old TOCTOU path (no lockf) still produces LAST_WRITER_WINS at w=4. The lockf fix is the actual delta. No regression introduced — the legacy path is intentionally left unfixed (it's the BEFORE state).

****Track 2 Verdict: PASS**
(contrast confirmed, no regression of fixed path)**

Track 3 — Deprecation Warn Hooks

Track 3a — pre-dispatch-gate.sh (P1-4)

****Command:**** `echo
'{"tool_name":"Task","tool_input":
{"prompt":"... MC #99999"}}' |
bash pre-dispatch-gate.sh`
****Fixture:**** Legacy

```
`/tmp/mehanik-cleared-99999`  
placed, no session-scoped path,  
`CLAUDE_SESSION_ID` unset  
**Observed stderr:** `[pre-  
dispatch-gate] DEPRECATION  
WARN: mehanik-cleared-99999  
found at legacy flat path  
/tmp/mehanik-cleared-99999.  
Two concurrent sessions on  
same MC both accept this —  
potential double-dispatch.  
Migrate to session-scoped path.`  
**Exit code:** 0 (hook continued,  
subsequent gate fired for  
unrelated probe reason —  
backward compat preserved)  
**DEPRECATION_WARN_COU
```

NT:** 1

****Track 3a Verdict: PASS****

Track 3b — schema-stub-gate.sh (P1-6)

****Command:**** `echo

'{"tool_name":"Bash","tool_input":{"command":"node mc.js ready 88888"}}' | bash schema-stub-gate.sh`

****Fixture:**** Legacy `/tmp/claim-schema-88888.json` placed, no session-scoped path

****Observed stderr:**** `[schema-stub-gate] DEPRECATION

WARN: claim-schema-88888.json found at legacy flat path /tmp/claim-schema-88888.json. Two sessions on same MC ID share this file. Migrate to session-scoped path.`
Exit code: 0 (backward compat preserved)
DEPRECATION_WARN_COUNT: 1

Track 3b Verdict: PASS

Track 3c — session-output-validator.sh (P1-5)

Command: Synthetic JSONL

transcript with legacy

```
`/tmp/evidence-77777/` path in  
John's message, fed to hook via  
`{"session_id":..., "transcript_path  
":...}` stdin
```

****Fixture:**** Legacy

```
`/tmp/evidence-  
77777/verification.json` with  
mtime past grandfather epoch  
(2026-05-18T01:00 > cutoff  
2026-05-11T17:15)
```

****Observed stderr:**** `[session-
output-validator]

DEPRECATION WARN: legacy
evidence dir /tmp/evidence-
77777/ used. Two sessions may
create same numeric dir. Migrate

to /tmp/alai//evidence-/.`

****Exit code:**** 0 (validation
SCORE=100, VIOLATIONS=0,
ACTION=none — backward
compat preserved)

****DEPRECATION_WARN_COU
NT:**** 1

****Track 3c Verdict: PASS****

**## Track 4 — Append-Path
Safety (P1-2, P1-3)**

****Pattern verification (grep on
mc.js):****

- `evidence-ledger.jsonl`:
`fs.openSync(ledgerPath, 'a')` +
`fs.writeFileSync` + `fs.fsyncSync` +
`fs.closeSync` (mc.js:329-334)
— O_APPEND with fsync,
single-write per entry

- `evidence-index.jsonl`:
`fs.appendFileSync(indexPath,
...)` (mc.js:226) — O_APPEND,
single JSON line per call

****Concurrent write test:**** 4
parallel Node.js processes
writing simultaneously against
tmp clones.

| File | Pre | Post | Expected |

Invalid JSON | Verdict |

|-----|-----|-----|-----

|-----|-----|

| evidence-ledger.jsonl | 2 | 6 | 6 |

0 | SAFE |

| evidence-index.jsonl | 1 | 5 | 5 |

0 | SAFE |

No truncation, no interleaved partial lines detected. PIPE_BUF atomicity (?512 bytes per entry) maintained.

CodeCraft's "VERIFIED-NO-CHANGE" status for P1-2 and P1-3 is confirmed. No regression-needed flag.

****Track 4 Verdict: SAFE (both
append paths)****

**## Track 5 — MEMORY.md
Daemon Serialisation (P1-12)**

****Daemon status check:****

...

node memory-writer.js status
? daemon: RUNNING | socket:
/tmp/alai/memory-writer.sock |
pid: 82720

...

****Serialisation test:** Inline test**

daemon started with socket at
`/tmp/t10-quad-track5-
81233/test-memory-writer.sock`,
writing to `/tmp/t10-quad-track5-
81233/MEMORY-clone.md`
(production MEMORY.md NOT
touched).

****4 concurrent `Promise.all`
append calls result:****

- All 4 responses:

`{"ok":true,"op":"append","bytes":
58}`

- Pre line count: 2, Post line
count: 6 (expected 6)

- Writer hits per line: [1, 1, 1, 1]
— each writer's line appears

exactly once, no interleave

-

```
`ALL_WRITERS_EXACTLY_ON  
E: true`
```

Note: Test used a tmp-clone daemon with identical serialisation queue logic from production memory-writer.js. Production MEMORY.md was not modified.

Track 5 Verdict: SAFE
(VERIFIED-PARTIAL note:
tested via tmp clone daemon,
not live production socket —
production daemon confirmed

RUNNING at pid 82720)**

Track 6 — DEFERRED Items
Spot-Check

P1-7 (Hop-build markers —
DEFERRED-requires-CAS-at-
dispatch-layer)

Grep: `grep -n "hop-build-
started" ~/system/kernel/pi-
orchestrator.js`

Result: Line 4028:

`fs.writeFileSync('/tmp/hop-build-
started-\${task.id}', ...)`

****Observation:**** Path is
`/tmp/hop-build-started-
\${task.id}` — per-task-id, not
per-session. The marker is task-
scoped, so the collision vector is
two sessions dispatching the
same task simultaneously. A
lockf on the file path cannot
prevent this — the race is at
task-dispatch decision level. Fix
requires CAS at task-start in
mc.js, not a file-path fix.

****Deferral reason: HONEST****

P1-10 (MCP Playwright —
DEFERRED-requires-MCP-side-
fix)

```
**Grep:** `grep -n "playwright"  
~/.claude/settings.json`
```

```
**Result:** Line 21:
```

```
`"mcp__playwright__*"` in allow  
list; Line 327: matcher for MCP  
tool. No hook file wraps MCP  
dispatch before the Playwright  
server — confirmed by `grep -rn  
"playwright" ~/.claude/hooks/  
returning only `settings.json:21`.
```

```
**Observation:** Playwright is an  
out-of-process singleton  
spawned by Claude Code  
runtime. There is no hook  
intercept point before MCP tool  
calls. Session isolation requires  
MCP-side browser context
```

implementation.

****Deferral reason: HONEST****

**### P1-13 (MC active-task
pointer — DEFERRED-probed-
SAFE-T3)**

****Grep:**** `sed -n '72,84p'
~/.claude/hooks/session-task-
lock-gate.sh`

****Result:**** Lines 75-81 contain
explicit comment: `# /tmp/mc-
active-task is single-writer,
world-writable, shared across all
sessions and daemons ? cross-
session contamination. Global
lock as shared mutable state in

concurrent system = design flaw,
not partial problem. Per-PPID
and per-PID markers are now
the ONLY authoritative blocking
source.`

****Observation:**** The design flaw
is explicitly acknowledged in
code. T3 probed SAFE at both
w=2 and w=4. The world-
writable global file is read for
audit/debug only (line 82+), not
for enforcement. Deferred to
technical debt backlog.

****Deferral reason: HONEST****

****Track 6 Verdict: PASS — all 3
DEFERRED items confirmed**

honestly tagged**

Cumulative Phase 1+2+3
Score

| Tier | Count | Status |

|-----|-----|-----|

| P0 | 7 | SAFE (from T10-ter —
session_state, last_verdict,
ledger_hash, costs_db,
incident_mode, prompt_forge,
skill_registry_db) |

| P1 APPLIED | 6 | SAFE (this
report — P1-1 lockf, P1-4/5/6
deprecate-warn, P1-8/9 CAS-

mv) |
| P1 VERIFIED | 4 | SAFE (P1-2
evidence-ledger, P1-3 evidence-
index, P1-11 LightRAG
semaphore, P1-12
MEMORY.md daemon) |
| P1 DEFERRED | 3 | Confirmed
honest (P1-7 hop-build, P1-10
Playwright MCP, P1-13 mc-
active-task) |
| P2 | 8 | LOW — all 8 P2
resources confirmed low via
CodeCraft code inspection |

****Total P1 resolved this sprint:
10 of 13 (6 APPLIED + 4
VERIFIED). 3 DEFERRED with**

honest rationale.**

Evidence Paths and sha256s

File	sha256	Type
`/tmp/session-collision-20260518T201729/probe.jsonl`	`e7ef05546f806baada9bb6e49a37a4652038fd37320523d11638b1b28c3a63ae`	probe harness output (Track 1)
`/tmp/session-collision-20260518T201737/probe.jsonl`	`978ee43dac797a039720b431ef`	

63e929b7c078ef627045909992
1ead0ace85aa` | probe harness
output (Track 2 legacy contrast) |
| `/tmp/t10-quad-track3-pre-
dispatch-v2-stderr.txt` |
`39140f8597a95719ff8ed3769c2
5be4ca2da6e8d65e4ff0402d244
9bdabf6c32` | Track 3a stderr
capture |
| `/tmp/t10-quad-track3-schema-
v2-stderr.txt` |
`197227e8eda38968ca84d978b
0deff415526e5d8619fac555601
b53107f2a3e7` | Track 3b stderr
capture |
| `/tmp/t10-quad-track3-sov-v3-
stderr.txt` |

`42cc8c6fd8d463694bb0d09df75
4367ea9a4220107b24a854cf4a
b30b86e30a9` | Track 3c stderr
capture |
| `/tmp/t10-quad-track4-
63076/evidence-ledger.jsonl` |
`ad36ef7d0b3f15574c2cc39f830
61e972df27fce54e3160c0845fab
b97412fdd` | Track 4 append
fixture (ledger) |
| `/tmp/t10-quad-track4-
63076/evidence-index.jsonl` |
`53211b28a932e8c68858b1891
7eec9eca306c46acf3d398fec77
6e6d485349cc` | Track 4 append
fixture (index) |
| `/tmp/t10-quad-track5-

81233/MEMORY-clone.md` |
`0065f74d6687c8636082d39d91
4b9619f5b9a6ee1234ce1cf3237
2aaf0596c03` | Track 5
MEMORY clone post-state |

Proveo sub-agent (Angie Jones). No production state modified. All writes to /tmp/only.

Revision #2

Created 2026-05-18 18:35:49 UTC by John

Updated 2026-06-21 20:03:46 UTC by John