

Meta-Agent Loop

Meta-Agent Loop

Owner: AgentForge **Implemented:** 2026-04-17 (Hive Activation Phase 5 T11) **Script:**

`~/system/tools/meta-agent-loop.js` **Schedule:** `~/Library/LaunchAgents/com.alai.meta-agent-loop.plist` — daily 03:30 CEST

Purpose

Before T11, nothing converted recurring lessons into new skills. If CodeCraft fixed the same class of bug three times, the third fix wasn't easier than the first.

The meta-agent loop reads HiveMind `learning` + `failed-task` intel from the last 24 hours, detects themes appearing 3+ times, and files a `NEW SKILL PROPOSAL` MC task for Alem to review.

Algorithm (intentionally simple)

1. Query intel in last 24h where `type IN ('learning', 'failed-task')`.
2. Tokenize each `message`, lowercase, drop stopwords + tokens shorter than 4 chars.
3. Build bigram frequency map — for each pair of consecutive tokens, count the distinct intel rows it appears in.
4. Take top 5 bigrams where distinct-row count ≥ 3 .
5. For each theme: check MC for an existing `NEW SKILL PROPOSAL: <theme>` (dedup across days). If absent, create.

Intentionally NOT doing: embedding clustering, LLM classification. Keep it legible; upgrade if bigram noise becomes real.

MC proposal shape

- Title: `NEW SKILL PROPOSAL: <theme bigram>`
- Owner: `skill-creator`
- Priority: `L`
- Description: theme, occurrence count, 3 sample learnings, draft SKILL.md template

Approval gate — hard rule

The loop **never** commits or pushes skills. Only proposes in MC. Alem reviews, approves, then a human (or skill-creator agent) runs `node ~/system/tools/library.js push <skill>`. This is deliberate: meta-learning without human-in-the-loop is how systems drift.

Manual run

```
node ~/system/tools/meta-agent-loop.js 2>&1 | tail -20
```

Idempotent. Re-running the same day won't create duplicates (MC dedup check).

Tuning knobs (in the script)

- `WINDOW_HOURS` — default 24. Raise to 48 if signal is sparse.
- `MIN_OCCURRENCES` — default 3. Raise to 5 if false positives.
- `TOP_N` — default 5. Cap on proposals per day.
- Stopword list — tune if common bigrams dominate (e.g. "task completed").

Known issues

- Bigrams alone produce some noise ("docker container", "container restart" both fire for the same lesson). Acceptable while we learn what themes matter.
- If the `skill-creator` owner isn't yet a real agent, proposals pile up unassigned. That's the designed state until we build one.

Related

- Failed-task signal comes from T12 — see `runbooks/mc-done-auto-writeback.md` for the failure regex.
- Learning entries come from T7 auto-writeback + manual `/learning-opportunity` invocations.
- Cross-link: `runbooks/hivemind-subscriptions.md` for how subscribers fire.

Revision #4

Created 2026-04-16 23:03:39 UTC by John

Updated 2026-07-26 20:03:10 UTC by John