$ claude --agent marcus

Persistent Agents on Claude Code

Nine always-on personal agents, eight months in daily use, and no agent framework

Dave Remy · Working note · 30 October 2026 · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Claude Code, tmux and systemd are enough

9
named agents, one per domain

8
months in daily use

227
scheduled jobs in one manifest

0
agent frameworks in between

Each agent is an ordinary claude --agent session in a tmux window, kept alive by systemd, reachable through its own Telegram bot.

Persistent Agents on Claude Code · Dave Remy · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Why build it: OpenClaw first, then Claude Code

PROBLEM 1

A separate identity

Its own name, Gmail and GitHub meant parallel accounts to keep in step. Work without value.

PROBLEM 2

Terms of service

Anthropic’s terms for using Claude through OpenClaw kept changing.

PROBLEM 3

Time

Real work still needed Claude Code on the side. Most time went to OpenClaw itself.

26 Feb LifeOS starts as a Claude Code project

24 Mar agents become persistent sessions

May nine agents

Aug move to a Linux server

Persistent Agents on Claude Code · Dave Remy · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Nine agents, one domain and one Telegram bot each

Marcus · Sonnet
Chief of staff. Routes and records every cross-domain message

Huberman · Opus
Health. Meals, workouts, sleep, labs, training plans

Buffett · Opus
Finance. Transactions, budgets, subscriptions

Franklin · Opus
Communications. Email, iMessage, contacts; never sends unapproved

Karpathy · Opus
CTO. Plans and runs software work as Agent Scrum

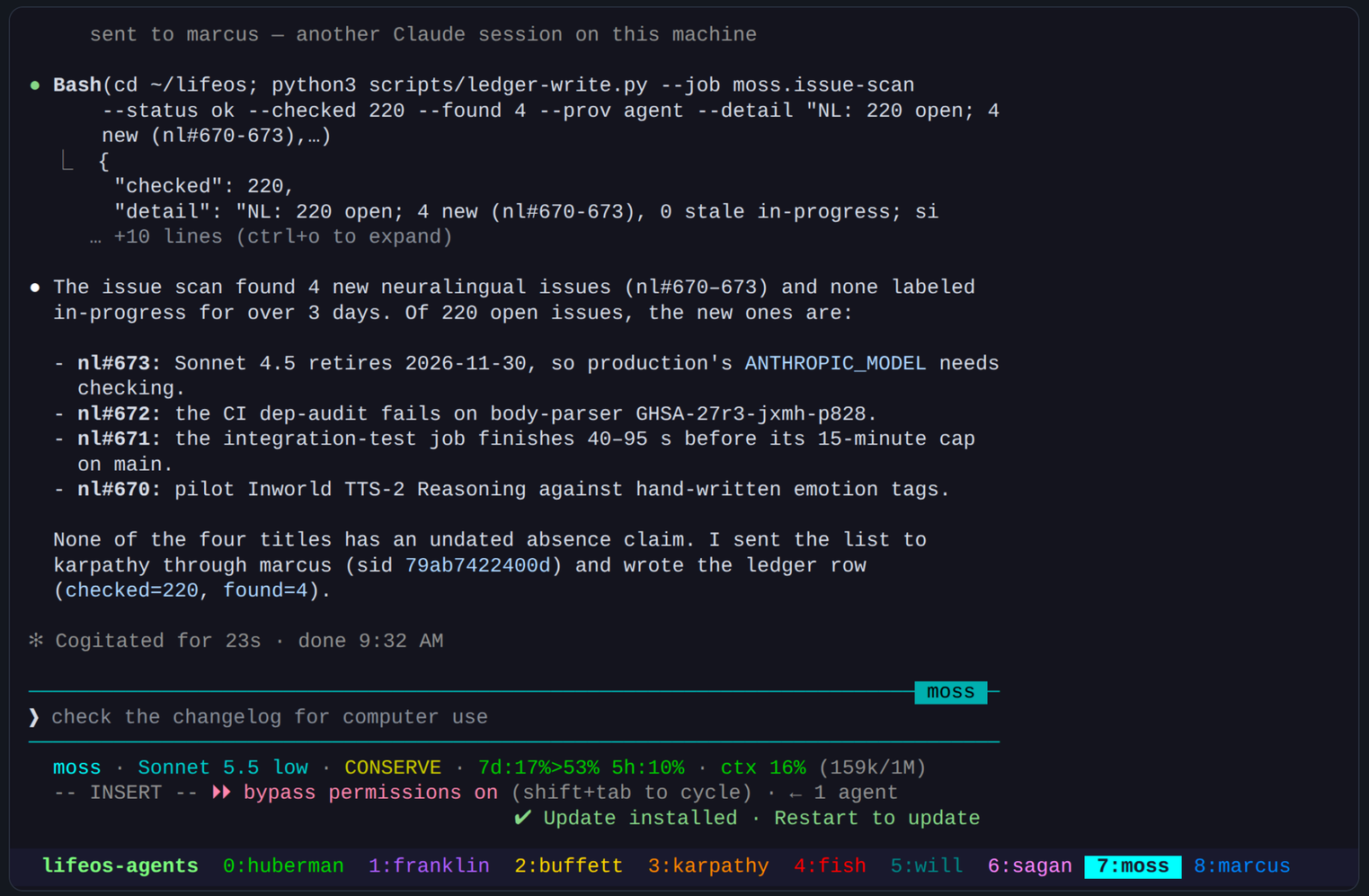
Moss · Sonnet
Operations. Watches the hub, its services and its jobs

Will · Opus
Customer support for Neuralingual

Sagan · Opus
Research. On request, plus a twice-weekly AI news scan

Fish · Opus
Marketing. Content and distribution for Neuralingual

Made in collaboration with AI



LIVE, THIS MORNING

One window, one agent

Moss ran its issue scan, sent the result to karpathy through marcus, and wrote a ledger row.

Bottom: the status line with model, budget state and usage, and the tmux bar with all nine agents.

Made in collaboration with AI

Anatomy of one agent

`systemd`	creates the tmux session at boot
`tmux window`	one per agent; attach from any device
`agent-wrapper.sh`	restart loop: model, pinned session, Telegram bot, time zone
`claude --agent name`	an ordinary Claude Code session
`name.md + CLAUDE.md`	role, voice, rules, what needs approval; in git

44
skills any agent can run

29
hooks, the strongest rules: they apply whether or not the model remembers

Scoped MCP
finance tools for finance, search tools for research

Persistent Agents on Claude Code · Dave Remy · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Four layers keep the agents running

`1 · systemd`

A user service creates the session at boot. Hub restarts go through it, never from inside a pane.

`2 · launcher`

Opens one window per agent, seconds apart. Nine sessions refreshing one login token at once logged them out.

`3 · wrapper`

A restart loop that resumes the agent’s pinned session and re-reads its definition.

`4 · watchdog`

Every five minutes, checks each window and restarts only the dead ones.

Two outages in July

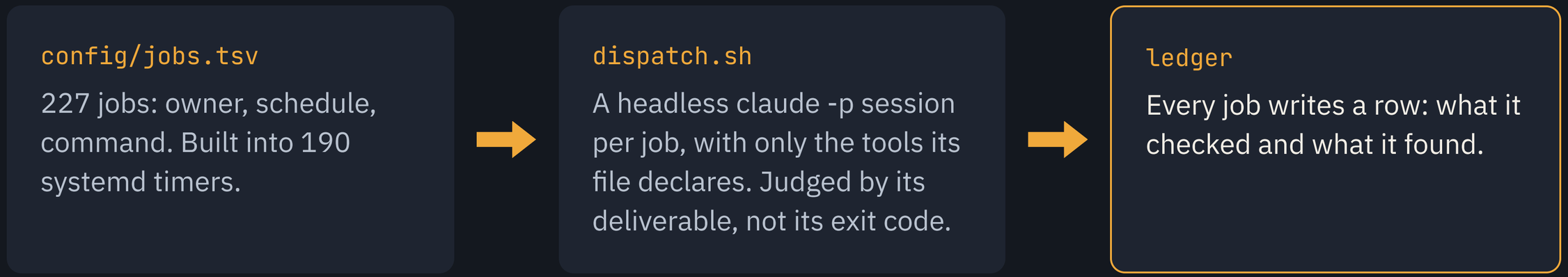
The whole hub went down twice and nothing brought it back. The watchdog exists because of those two outages.

Persistent Agents on Claude Code · Dave Remy · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Scheduled work runs outside the sessions

Claude Code’s own scheduled tasks expire after seven days, and a run missed while the session is down is never made up.



Without the ledger, a job that ran and found nothing looked the same as a job that never ran.

Persistent Agents on Claude Code · Dave Remy · doi:10.5281/zenodo.23787645

Made in collaboration with AI

Messages and memory

Two kinds of traffic

Routine questions
go directly from agent to agent with SendMessage

Cross-domain signals
go through the chief of staff, which records each in a daily signals file and forwards it

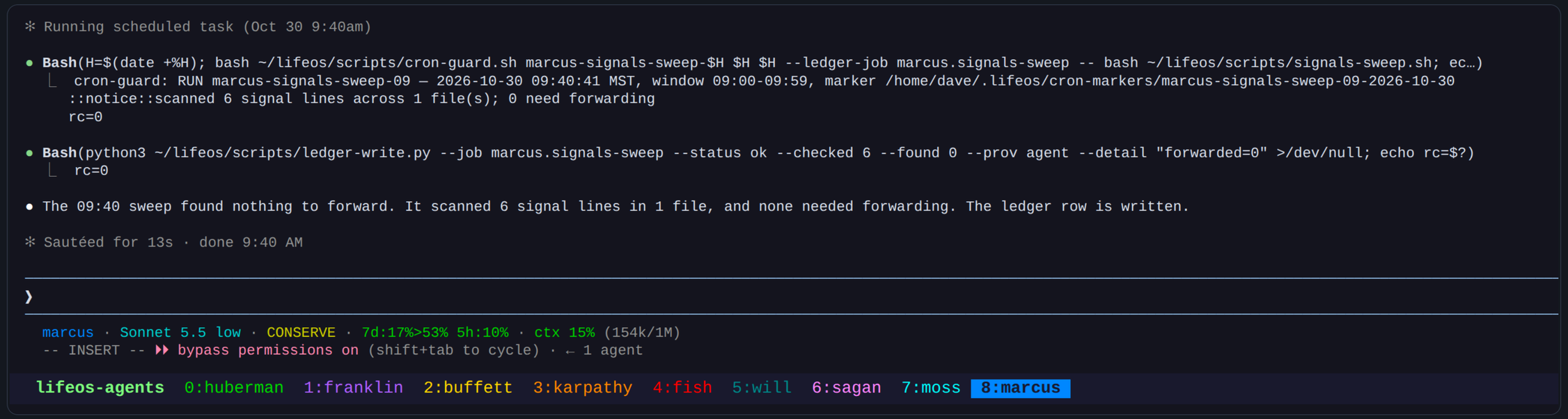
Sender-minted IDs
so a retry is recognised, not delivered twice

Five stores, one kind of fact each

Store	Holds
Definitions, CLAUDE.md	Identity and protocol
Obsidian vault	Durable knowledge, daily notes
Auto-memory	Lessons across sessions
Handoff documents	In-flight work across restarts
Git, signals, ledger, ticklers	Events, append-only

Made in collaboration with AI

The router, between messages



Marcus’s hourly sweep looks for signals that were appended to the file because the router couldn’t be reached. This one found nothing to forward and still wrote its ledger row, so “found nothing” is on record.

Captured from the hub’s tmux session, 30 October 2026, 09:40

Made in collaboration with AI

What persistence exposed, and what we added

Problem	What we added
A restart loses the agent’s working state	Handoff document per agent; drain before handoff; pinned session IDs
In-session tasks expire after 7 days	Job manifest compiled to systemd timers; headless jobs
Nine agents messaging with no record	Router agent and daily signals file; message IDs
“Found nothing” looks like “never ran”	Ledger row per job: checked and found
Phone buzzing while at the terminal	Presence check on attached tmux clients
Rate limits shared by nine sessions	Budget state file moves the fleet between models
One login token for all sessions	Staggered start; scripted account switch
Real access to email, money, messages	Hooks block sends; written approval list

Made in collaboration with AI

Failures that changed the design

2026	What happened	What changed
July	Hub died twice and nothing restarted it	Per-window watchdog
12 Aug	Haiku agents wrote replies into their own pane; no one got them	No Haiku anywhere
14 Aug	A guard skipped correctly and silently; read as an outage	The ledger
19 Aug	UTC clock: a 07:26 reminder would fire at 00:26	Wrapper sets the time zone
21 Aug	Publish reported success; the API refused every request	Count it only when read back
29 Aug	Logout in one pane logged out all nine	Scripted account switch
9 Sep	An agent restarted the hub mid-conversation	CTO’s agreement and a notice first

Made in collaboration with AI

Most of these failures were silent.

A reply that reached no one, a job that skipped without a trace, a publish that reported success. Much of the infrastructure exists to make silent failures visible.

Made in collaboration with AI

Compared with hosted persistent agents

	This setup	OpenClaw	Grok Bot	Muse	Claude Duet
Released	Mar 2026, private	Jan 2026	Aug 2026, beta	Sep 2026	Oct 2026, preview
Runs on	My Linux server	Your machine	Cloud VM per user	Cloud VM per user	Cloud VM per user
Agents	Nine, own permissions each	One with skills	Several bots, one VM	One	One
Approvals	Hooks and rules we maintain	By configuration	Built in	Built in	Built in
Maintained by	Me and the agents	Me	xAI and Cursor	Meta	Anthropic
Edits own code	Yes, through git	Skills are files	No	No	No

Hosted products remove most of the infrastructure, and for most people they are the better choice. What they don’t offer is the integration: the agents that run my email and money share a repository with the agents that ship my software.

Details for hosted products from their documentation and press, October 2026 · Claude Duet is a research preview

Made in collaboration with AI

The cost: the infrastructure is real software

3,761
commits since 26 Feb, close to Neuralingual's 3,879

434
bash files, about 91,000 lines

3
review rounds on average for gate and parser changes

More of my development time than any one major project, less than all of them together. Shell scripts were the fastest way to add each piece and became the most expensive to maintain. In October we stopped writing them and began moving tooling to TypeScript.

Made in collaboration with AI

What is open

Maintenance
Weekly time spent on infrastructure is unmeasured

Overlap with Claude Duet
Untested; Duet is waitlisted

Single points of failure
One server, one shared login token, one approver

Remote use
The Claude Code desktop app is not yet part of it

`daveremy.github.io/persistent-agents`

Working note, PDF and source: github.com/daveremy/persistent-agents · doi:10.5281/zenodo.23787645 · CC BY 4.0 · Companion paper: Agent Scrum

Made in collaboration with AI