



OpenClaw

THE AI THAT ACTUALLY DOES THINGS.

Clears your inbox, sends emails, manages your calendar, checks you in for flights.

All from WhatsApp, Telegram, or any chat app you already use.

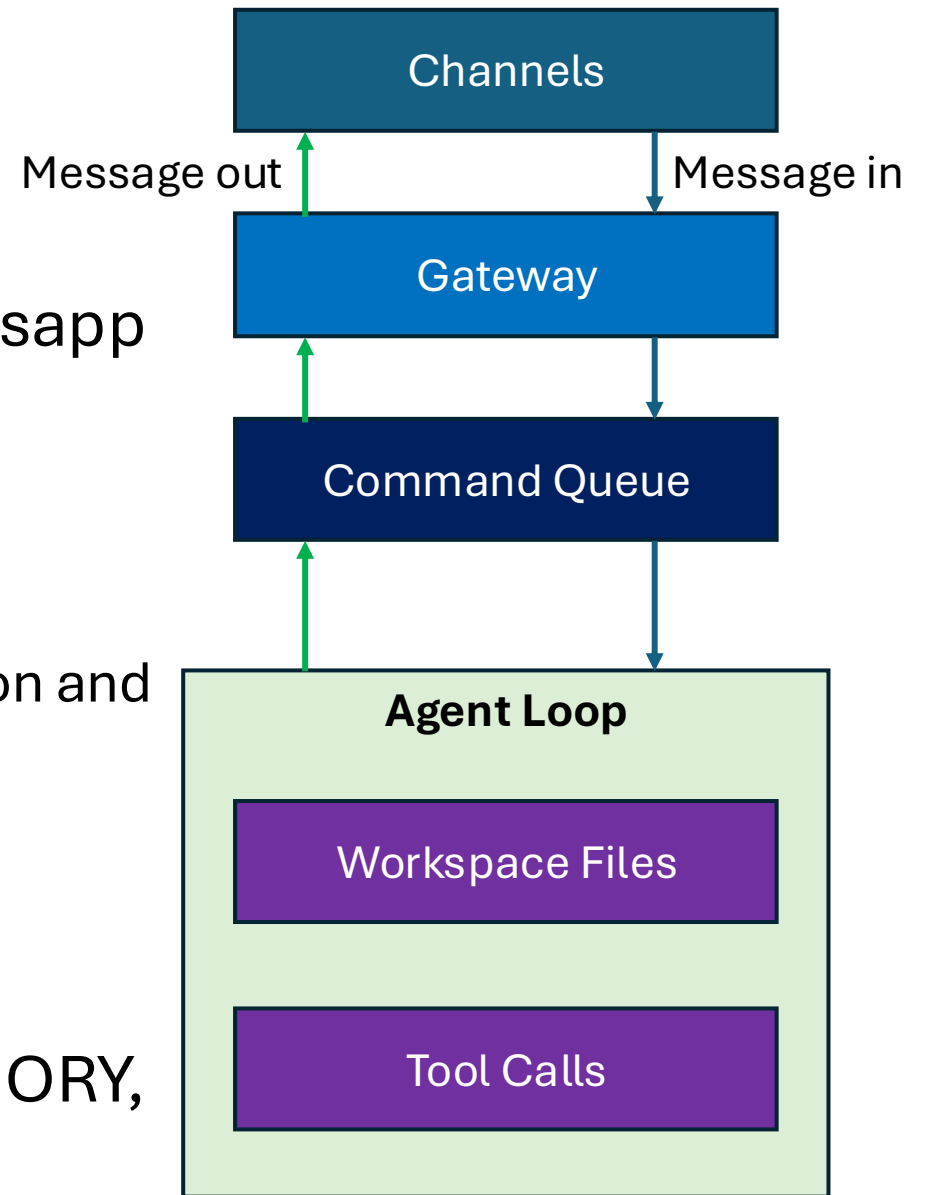
OpenClaw

- Always-on Personal AI Assistant
- Gateway (control plane) that connects chat channels to agent runtime and tools
 - Ease of use (lots of built in skills and integrations)
 - Talk to agent anywhere at anytime
 - Easy to self-host
- Only limited by what data you give OpenClaw access to.



Core Components

- Channels: Telegram / iMessage / Slack / Whatsapp / Discord / etc.
- Gateway: Ingestion, Auth, Scheduling, UI
 - Single long-lived process for sessions / routing.
- Command Queue:
 - “main” lane for inbound + heartbeats. Also has cron and subagent lanes
- Agent Loop: Intake, Context, Model, Tools, Persistence
 - Single agent workspace directory
 - Subagents only inject Agents.md + Tools.md.
- Workspace Files: SOUL, USER, AGENTS, MEMORY, etc.



Workspace Files

- SOUL: Persona / boundaries / tone
- USER: User Profile
- AGENTS: Operating Instructions / Memory Conventions
- HEARTBEAT: Tasks for Agent to check periodically
- MEMORY: Long-term Memory (short-term memory captured daily)
- TOOLS: Available Tools
- IDENTITY: Agent Identity Information

OpenClaw Configuration

- Terminal-based configuration setup
- Model access through any major provider or local models
- Channels galore

```
• Where will the Gateway run?
  Local (this machine)

• Select sections to configure
  Channels

• Channel status
  Telegram: configured
  WhatsApp (default): not linked
  Discord: needs token
  Slack: needs tokens
  Signal: needs setup
  signal-cli: missing (signal-cli)
  iMessage: needs setup
  imsg: found (imsg)
  IRC: not configured
  Google Chat: not configured
  Feishu: install plugin to enable
  Google Chat: install plugin to enable
  Nostr: install plugin to enable
  Microsoft Teams: install plugin to enable
  Mattermost: install plugin to enable
  Nextcloud Talk: install plugin to enable
  Matrix: install plugin to enable
  BlueBubbles: install plugin to enable
  LINE: install plugin to enable
  Zalo: install plugin to enable
  Zalo Personal: install plugin to enable
  Synology Chat: install plugin to enable
  Tlon: install plugin to enable
```

```
OpenClaw 2026.3.2 (85377a2) - The lobster in your shell.

• Doctor warnings
  - channels.telegram.groupPolicy is "allowlist" but groupAllowFrom (and allowFrom) is
    empty - all group messages will be silently dropped. Add sender IDs to
    channels.telegram.groupAllowFrom or channels.telegram.allowFrom, or set groupPolicy to
    "open".
  - channels.telegram.accounts.default.groupPolicy is "allowlist" but groupAllowFrom (and
    allowFrom) is empty - all group messages will be silently dropped. Add sender IDs to
    channels.telegram.accounts.default.groupAllowFrom or
    channels.telegram.accounts.default.allowFrom, or set groupPolicy to "open".
  - channels.telegram.accounts.goofy3bot.groupPolicy is "allowlist" but groupAllowFrom
    (and allowFrom) is empty - all group messages will be silently dropped. Add sender IDs
    to channels.telegram.accounts.goofy3bot.groupAllowFrom or
    channels.telegram.accounts.goofy3bot.allowFrom, or set groupPolicy to "open".

OPENCLAW

• OPENCLAW

OpenClaw configure

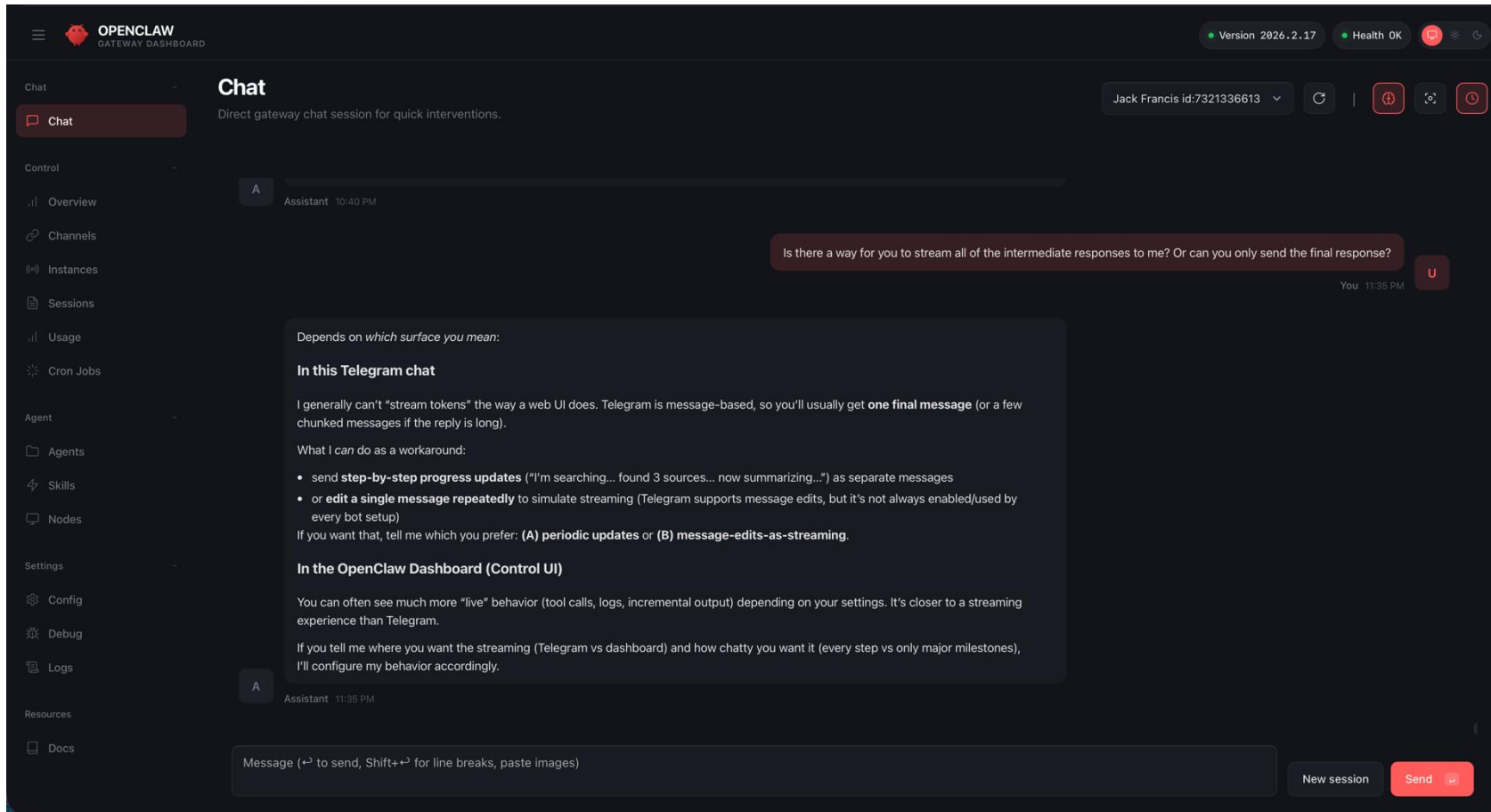
• Existing config detected
  workspace: ~/.openclaw/workspace
  model: openai-codex/gpt-5.2
  gateway.mode: local
  gateway.port: 18789
  gateway.bind: loopback

• Where will the Gateway run?
  Local (this machine)

• Select sections to configure
  Model

• Model/auth provider
  • OpenAI (Codex OAuth + API key)
    • Anthropic
    • Chutes
    • vLLM
    • MiniMax
    • Moonshot AI (Kimi K2.5)
    • Google
    • xAI (Grok)
    • Mistral AI
    • Volcano Engine
    • BytePlus
    • OpenRouter
    • Kilo Gateway
    • Qwen
    • Z.AI
    • Qianfan
    • Copilot
```

OpenClaw Dashboard

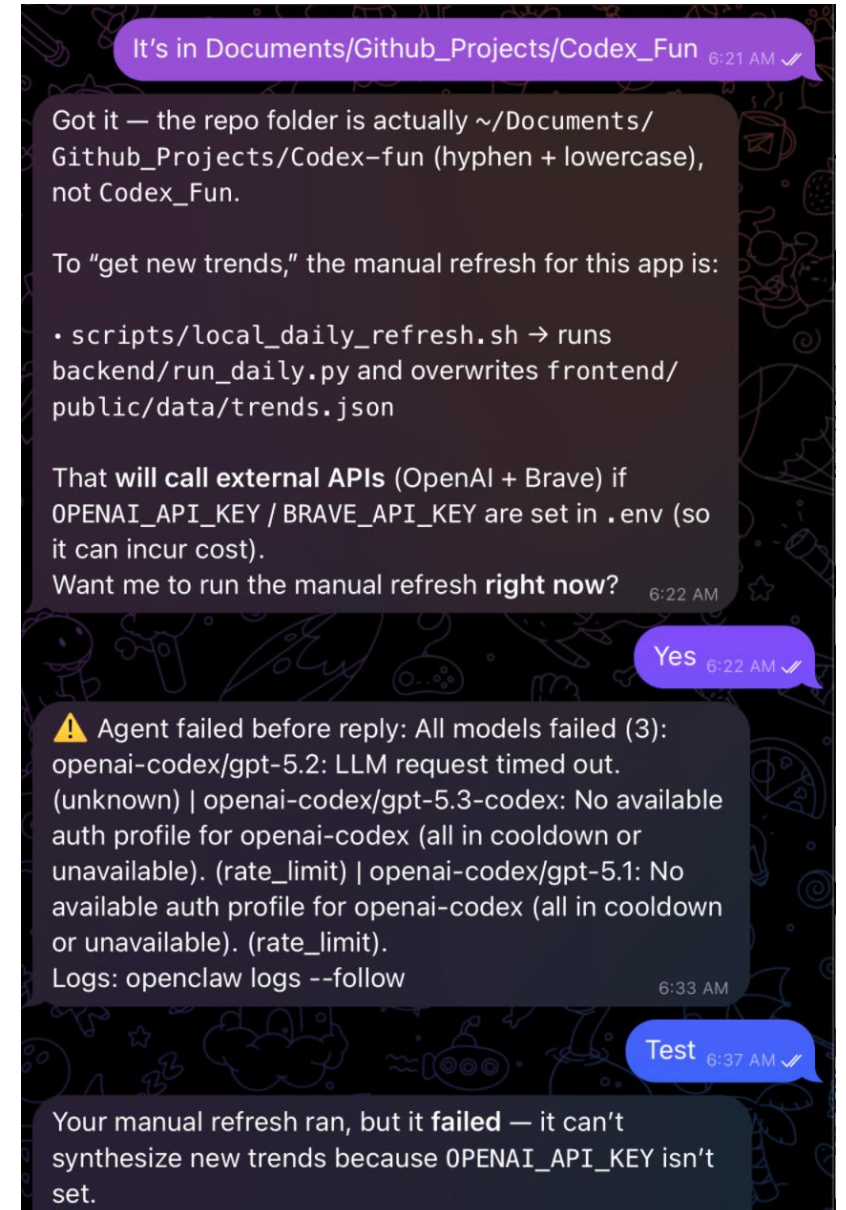


Things to Review in Dashboard

- Sessions > Verbosity
- Usage
- Cron Jobs vs Heartbeat
- Agents
 - Workspace Files and more
- Skills

Things I've Learned

- Strongly recommend Codex OAuth as your model if you choose to not run a local model
 - Very generous rate limits currently.
 - OpenClaw does much better with GPT5.2 over GPT5 mini.
- If you don't go with Codex OAuth and decide to use API keys, don't use the most powerful model for everything. You will burn money
 - Setup a smaller model for heartbeats and regularly change model being used in your channel interface
- Default is 4 concurrent main agents and 8 concurrent subagents. If all of these are Opus 4.6 and an issue happens, it's expensive



Things I've Learned (cont.)

- Default configurations for OpenClaw are relatively straightforward (probably 2 hours of setup time).
 - If you go off the beaten path, good luck. Some things are trivial, some are difficult.
 - Nice benefit -> OpenClaw understands OpenClaw well, so just ask it to fix
- Most people run OpenClaw headless, but you will need ssh access to box (and preferably screen sharing). Just makes life easier
- Permissions, Permissions, Permissions. Be careful about what you give OpenClaw access to.

OpenClaw Alternatives

- NanoClaw (4k LoC vs 434k, isolated containers: <https://nanoclawn.dev>)
- NanoBot (4k LoC vs 434k: <https://github.com/HKUDS/nanobot>)
- OpenFang (built in Rust: <https://www.openfang.sh>)
- PicoClaw (NanoBot in Go: <https://github.com/sipeed/picoclawn>)
- ZeroClaw (built in Rust: <https://github.com/zeroclawn-labs/zeroclawn>)

Benchmark Snapshot (ZeroClaw vs OpenClaw, Reproducible)

Local machine quick benchmark (macOS arm64, Feb 2026) normalized for 0.8GHz edge hardware.

	OpenClaw	NanoBot	PicoClaw	ZeroClaw 🦀
Language	TypeScript	Python	Go	Rust
RAM	> 1GB	> 100MB	< 10MB	< 5MB
Startup (0.8GHz core)	> 500s	> 30s	< 1s	< 10ms
Binary Size	~28MB (dist)	N/A (Scripts)	~8MB	~8.8 MB
Cost	Mac Mini \$599	Linux SBC ~\$50	Linux Board \$10	Any hardware \$10

3 Demos + Fun

- My use case and how I use OpenClaw
- Building an Application with OpenClaw
- Setting up Telegram Bot with OpenClaw
- Play around with OpenClaw

Telegram bot demo commands

- `openclaw channels add --channel telegram --account <acct_name> --token "PASTE_THE_BOTFATHER_TOKEN" --name <agent_name>`
- `openclaw config set channels.telegram.accounts.<acct_name>.botToken "PASTE_TOKEN_HERE"`
- `openclaw gateway restart`
- `openclaw channels status --probe`