































|   |   |
|---|---|
|  <b>viettranx</b> chore: apply go fix modernizations ✓ | 364d2d3 · 5 days ago  |
|  .github   | ci(release): native arm64 runners + split-bui... last week    |
|  _readmes  | feat(build): embed web UI in backend binar... last month      |
|  _statics  | feat(v3): core architecture redesign — pipel... 3 weeks ago   |
|  cmd   | fix(doctor): handle empty display_name (#1... 5 days ago      |
|  compose.d   | Add prepare-compose.sh for composable d... 2 months ago       |
|  compose.options                                       | Add prepare-compose.sh for composable d... 2 months ago       |
|  docker  | fix(docker): pin Python/npm dependency ve... last month       |
|  docs  | docs(changelog): v3.11.3 hotfix entry (#1034) 5 days ago      |
|  examples/hooks  | docs(hooks): user guide + example configs ... 2 weeks ago     |
|  internal  | chore: apply go fix modernizations 5 days ago                 |
|  migrations  | fix(providers): heartbeat FK SET NULL + te... 5 days ago      |
|  options/postgres                                    | Add prepare-compose.sh for composable d... 2 months ago       |
|  pkg   | feat(pipeline): session compaction overflow... 2 weeks ago    |
|  scripts   | test: speed up retry/cron/facebook tests, dr... 3 weeks ago   |
|  skills  | fix(security): replace stdlib xml.etree.Eleme... 2 months ago |
|  tests   | test: integration coverage for #1034 (provid... 5 days ago    |
|  ui  | feat(pancake): add TikTok sub-platform sup... last week       |
|  .dockerignore                                       | fix(docker): exclude node deps and dist fro... 2 weeks ago    |
|  .env.example  | fix(docker): restore base capabilities in san... 2 months ago |
|  .gitattributes                                      | fix: Windows deployment — broken symlink... 2 months ago      |
|  .gitignore  | chore: ignore nginx.staging.conf in ui/web 2 weeks ago        |
|  .repomixignore                                      | feat: Introduce .repomixignore for repomix... 2 months ago    |
|  CHANGELOG.md  | refactor(pancake): simplify private-reply to ... last week    |
|  CLAUDE.md   | ci(release): native arm64 runners + split-bui... last week    |
|  CONTRIBUTING.md                                     | ci: add layered test stages for hybrid test st... 3 weeks ago |
|  Dockerfile  | feat(packages): GitHub Releases binary ins... 2 weeks ago     |
|  Dockerfile.sandbox                                  | Initial commit: GoClaw AI agent gateway 3 months ago          |
|  LICENSE   | Update copyright holder 3 weeks ago                           |
|  Makefile  | fix(make): use --pull always in up, split up-b... 2 weeks ago |
|   | feat(v3): core architecture redesign — pipel... 3 weeks ago   |

|                                |  |              |
|--------------------------------|--|--------------|
| README.md                      |  |              |
| api-reference.md               | refactor: deprecate standalone mode, man...      | 2 months ago |
| docker-compose.browser.yml     | fix(docker): swap browser sidecar to chrom...    | 2 weeks ago  |
| docker-compose.claude-cli.yml  | fix(docker): restore base capabilities in san... | 2 months ago |
| docker-compose.otel.yml        | feat(zalo): add file and image upload/send ...   | 2 months ago |
| docker-compose.postgres.yml    | feat(build): embed web UI in backend binar...    | last month   |
| docker-compose.redis.yml       | feat(cache): add build-tag-gated Redis cac...    | 2 months ago |
| docker-compose.sandbox.yml     | fix(security): harden sandbox, auth, and sh...   | last month   |
| docker-compose.selfservice.yml | feat(docker): make goclaw-ui stop instantly ...  | last month   |
| docker-compose.tailscale.yml   | refactor: remove standalone mode, consoli...     | 2 months ago |
| docker-compose.upgrade.yml     | Add prepare-compose.sh for composable d...       | 2 months ago |
| docker-compose.yml             | feat(build): embed web UI in backend binar...    | last month   |
| docker-entrypoint.sh           | feat(packages): GitHub Releases binary ins...    | 2 weeks ago  |
| go.mod                         | refactor(pancake): simplify private-reply to ... | last week    |
| go.sum                         | refactor(pancake): simplify private-reply to ... | last week    |
| main.go                        | feat: Implement timezone support for cron s...   | 3 months ago |
| prepare-compose.sh             | Add prepare-compose.sh for composable d...       | 2 months ago |
| prepare-env.sh                 | feat(build): embed web UI in backend binar...    | last month   |
| websocket-protocol.md          | Initial commit: GoClaw AI agent gateway          | 3 months ago |

README Contributing License



### Multi-Tenant AI Agent Platform

Multi-agent AI gateway built in Go. 20+ LLM providers. 7 channels. Multi-tenant PostgreSQL. Single binary. Production-tested. Agents that orchestrate for you.

[Documentation](#) • [Quick Start](#) • [Twitter / X](#)

Go 1.26
PostgreSQL 18
Docker
WebSocket
OpenTelemetry
Anthropic
OpenAI Compatible
License
CC BY-NC 4.0

**Languages:**
[简体中文](#) · 
 [日本語](#) · 
 [한국어](#) · 
 [Tiếng Việt](#) · 
 [Tagalog](#) · 
 [Español](#) · 
 [Português](#) · 
 [Italiano](#) · 
 [Deutsch](#) · 
 [Français](#) · 
 [العربية](#) · 
 [हिन्दी](#) · 
 [Русский](#) · 
 [বাংলা](#) · 
 [עברית](#) · 
 [Polski](#) · 
 [Čeština](#) · 
 [Nederlands](#) · 
 [Türkçe](#) · 
 [Українська](#) · 
 [Bahasa Indonesia](#) · 
 [ไทย](#) · 
 [اردو](#) · 
 [Română](#) · 
 [Svenska](#) · 
 [Ελληνικά](#) · 
 [Magyar](#) · 
 [Suomi](#) · 
 [Dansk](#) · 
 [Norsk](#)

## Core Features

- **8-Stage Agent Pipeline** — context → history → prompt → think → act → observe → memory → summarize. Pluggable stages, always-on execution
- **4-Mode Prompt System** — Full / Task / Minimal / None with section gating, cache boundary optimization, and per-session mode resolution
- **3-Tier Memory** — Working (conversation) → Episodic (session summaries) → Semantic (knowledge graph). Progressive loading L0/L1/L2
- **Knowledge Vault** — Document registry with [[wikilinks]], hybrid search (FTS + pgvector), filesystem sync
- **Agent Teams & Orchestration** — Shared task boards, inter-agent delegation (sync/async), 3 orchestration modes (auto/explicit/manual)
- **Self-Evolution** — Metrics → suggestions → auto-adapt with guardrails. Agents refine their own communication style
- **Multi-Tenant PostgreSQL** — Per-user workspaces, per-user context files, encrypted API keys (AES-256-GCM), RBAC, isolated sessions
- **20+ LLM Providers** — Anthropic (native HTTP+SSE with prompt caching), OpenAI, OpenRouter, Groq, DeepSeek, Gemini, Mistral, xAI, MiniMax, DashScope, Claude CLI, Codex, ACP, and any OpenAI-compatible endpoint
- **7 Messaging Channels** — Telegram, Discord, Slack, Zalo OA, Zalo Personal, Feishu/Lark, WhatsApp
- **Production Security** — 5-layer permission system, rate limiting, prompt injection detection, SSRF protection, AES-256-GCM encryption
- **Single Binary** — ~25 MB static Go binary, no Node.js runtime, <1s startup, runs on a \$5 VPS
- **Observability** — Built-in LLM call tracing with spans and prompt cache metrics, optional OpenTelemetry OTLP export

## Desktop Edition (GoClaw Lite)

A native desktop app for local AI agents — no Docker, no PostgreSQL, no infrastructure.

### macOS:

```
curl -fsSL https://raw.githubusercontent.com/nextlevelbuilder/goclaw/main/scripts/install-lite.sh | bash
```

### Windows (PowerShell):

```
irm https://raw.githubusercontent.com/nextlevelbuilder/goclaw/main/scripts/install-lite.ps1 | iex
```

## What's Included

- Single native app (Wails v2 + React), ~30 MB
- SQLite database (zero setup)
- Chat with agents (streaming, tools, media, file attachments)
- Agent management (max 5), provider config, MCP servers, skills, cron
- Team tasks with Kanban board and real-time updates
- Auto-update from GitHub Releases

## Lite vs Standard

| Feature             | Lite (Desktop)    | Standard (Server)                                |
|---------------------|-------------------|--|
| Agents              | Max 5             | Unlimited  |
| Teams               | Max 1 (5 members) | Unlimited  |
| Database            | SQLite (local)    | PostgreSQL                                       |
| Memory              | FTS5 text search  | pgvector semantic                                |
| Channels            | —                 | Telegram, Discord, Slack, Zalo, Feishu, WhatsApp |
| Knowledge Graph     | —                 | Full   |
| RBAC / Multi-tenant | —                 | Full   |
| Auto-update         | GitHub Releases   | Docker / binary                                  |

### Building from Source

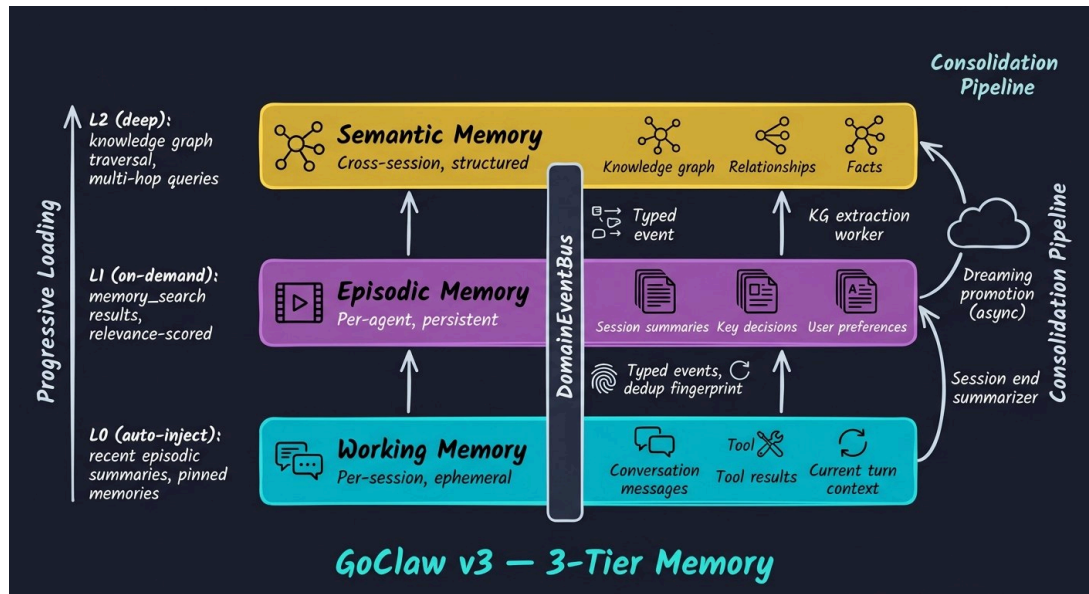
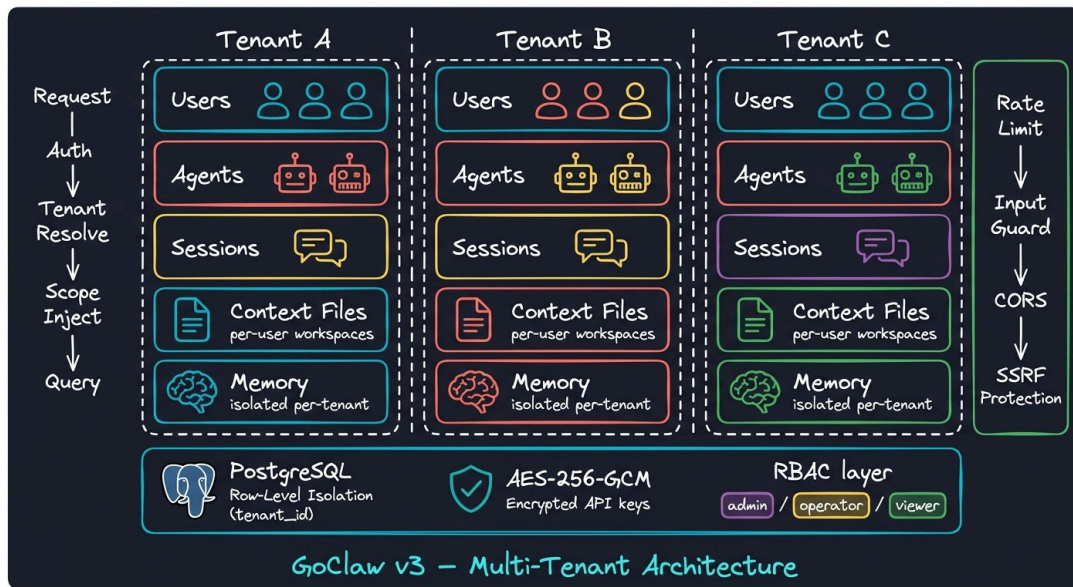
```
# Prerequisites: Go 1.26+, pnpm, Wails CLI (go install github.com/wailsapp/wails/v2/cmd/wails@latest)
make desktop-build           # Build .app (macOS) or .exe (Windows)
make desktop-dmg VERSION=0.1.0 # Create .dmg installer (macOS only)
make desktop-dev             # Dev mode with hot reload
```

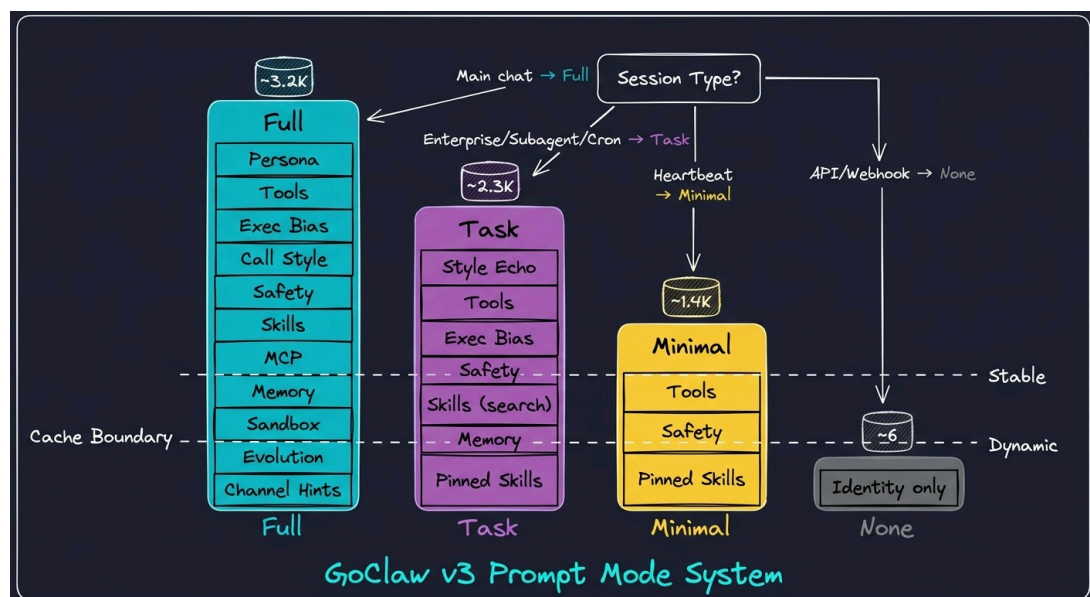
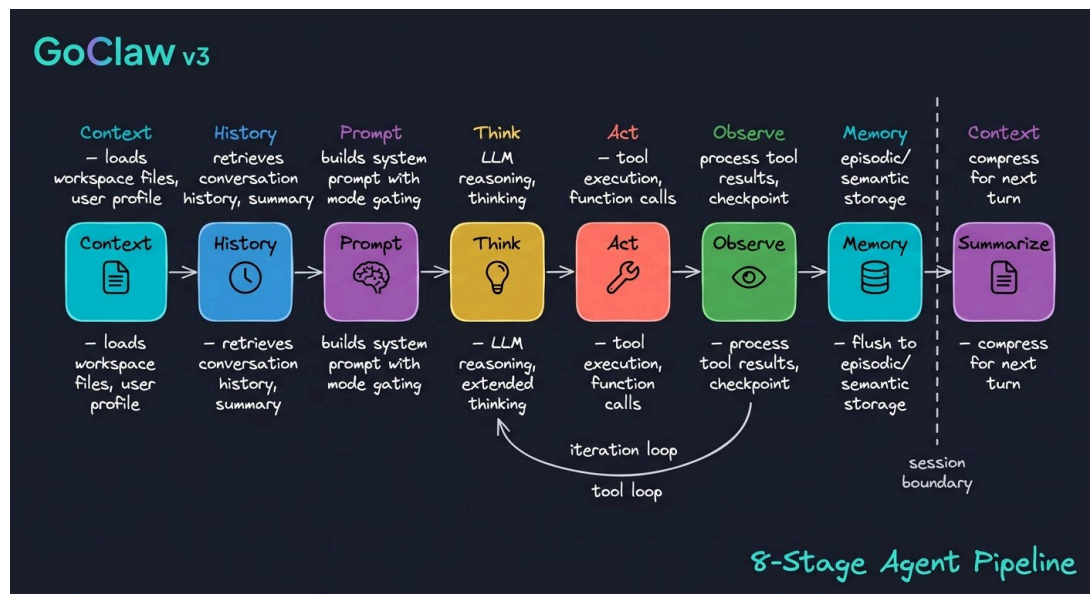
### Desktop Releases

Desktop uses independent versioning with `lite-v*` tags:

```
git tag lite-v0.1.0 && git push origin lite-v0.1.0
# -> GitHub Actions builds macOS (.dmg + .tar.gz) + Windows (.zip)
# -> Creates GitHub Release with all assets
```

### Architecture





## Quick Start

**Prerequisites:** Go 1.26+, PostgreSQL 18 with pgvector, Docker (optional)

## From Source

```
git clone -b main https://github.com/nextlevelbuilder/goclaw.git && cd goclaw
make build
./goclaw onboard # Interactive setup wizard
source .env.local && ./goclaw
```

**Note:** The default branch is `dev` (active development). Use `-b main` to clone the stable release branch.

## With Docker

```
# Generate .env with auto-generated secrets
chmod +x prepare-env.sh && ./prepare-env.sh

# Add at least one GOCLAW*_API_KEY to .env, then:
make up

# Web Dashboard at http://localhost:18790 (built-in)
```

```
# Health check: curl http://localhost:18790/health

# Optional: separate nginx for custom SSL/reverse proxy
# make up WITH_WEB_NGINX=1 → Dashboard at http://localhost:3000
```

`make up` creates a Docker network, embeds the correct version from git tags, builds and starts all services, and runs database migrations automatically.

#### Common commands:

```
make up           # Start all services (build + migrate)
make down        # Stop all services
make logs        # Tail logs (goclaw service)
make reset       # Wipe volumes and rebuild from scratch
```

**Optional services** — enable with `WITH_*` flags:

| Flag                          | Service         | What it does  |
|-------------------------------|-----------------|---|
| <code>WITH_BROWSER=1</code>   | Headless Chrome | Enables <code>browser</code> tool for web scraping, screenshots, automation |
| <code>WITH_OTEL=1</code>      | Jaeger          | OpenTelemetry tracing UI for debugging LLM calls and latency                |
| <code>WITH_SANDBOX=1</code>   | Docker sandbox  | Isolated container for running untrusted code from agents                   |
| <code>WITH_TAILSCALE=1</code> | Tailscale       | Expose gateway over Tailscale private network                               |
| <code>WITH_REDIS=1</code>     | Redis           | Redis-backed caching layer  |

Flags can be combined and work with all commands:

```
# Start with browser automation and tracing
make up WITH_BROWSER=1 WITH_OTEL=1

# Stop everything including optional services
make down WITH_BROWSER=1 WITH_OTEL=1
```

When `GOCLAW_*_API_KEY` environment variables are set, the gateway auto-onboards without interactive prompts — detects provider, runs migrations, and seeds default data.

#### Docker image variants:

| Image                    | Description   |
|--------------------------|---|
| <code>latest</code>      | Backend + embedded web UI + Python ( <b>recommended</b> ) |
| <code>latest-base</code> | Backend API-only, no web UI, no runtimes                  |
| <code>latest-full</code> | All runtimes + skill dependencies pre-installed           |
| <code>latest-otel</code> | Latest + OpenTelemetry tracing                            |
| <code>goclaw-web</code>  | Standalone nginx + React SPA (for custom reverse proxy)   |

For custom builds (Tailscale, Redis): `docker build --build-arg ENABLE_TSNET=true ...` See the [Deployment Guide](#) for details.

## Updating

### Docker

```
docker compose pull && docker compose up -d
```

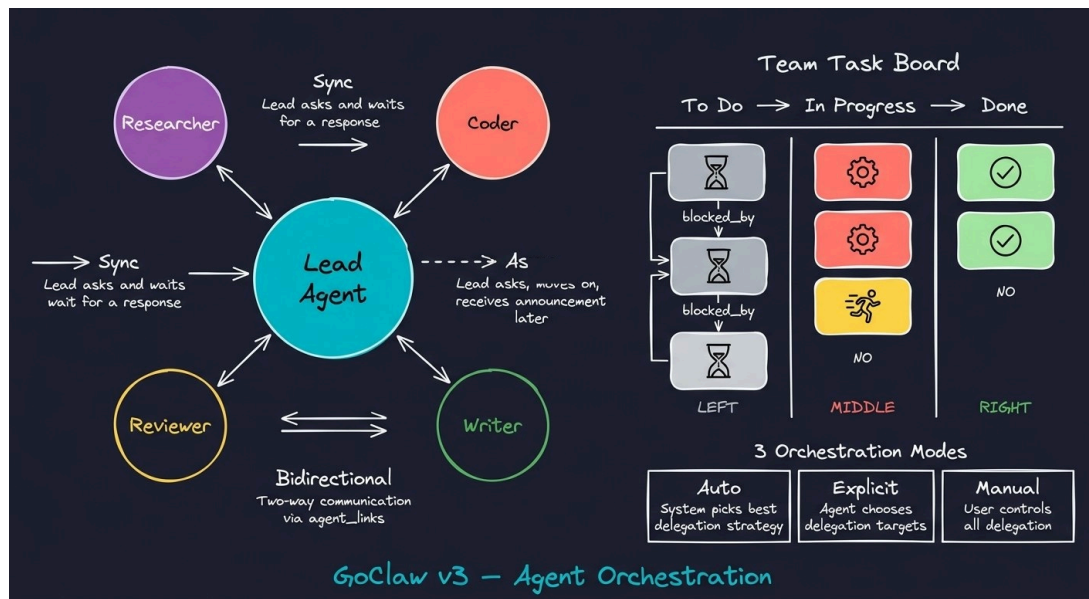
### Binary (with embedded web UI)

```
goclaw update --apply # Downloads, verifies SHA256, swaps binary, restarts
```

### Web Dashboard

Open **About** dialog → click **Update Now** (admin only). The update includes both backend and web dashboard when using the default **latest** image.

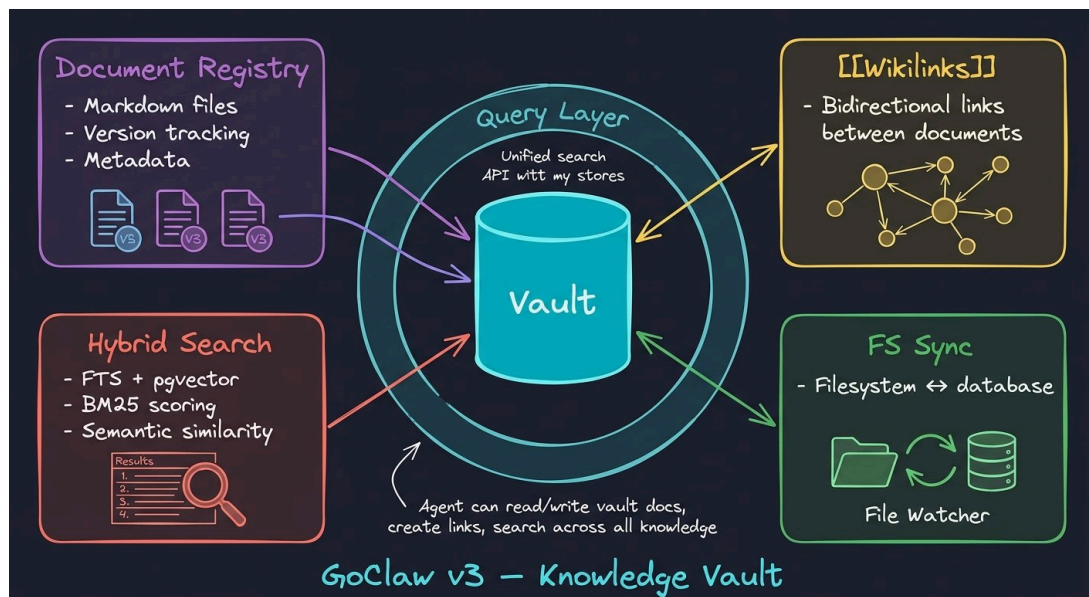
### Multi-Agent Orchestration



Each agent runs with its own identity, tools, LLM provider, and context files. Three delegation modes — sync (wait), async (fire-and-forget), bidirectional — connected through explicit permission links with concurrency limits.

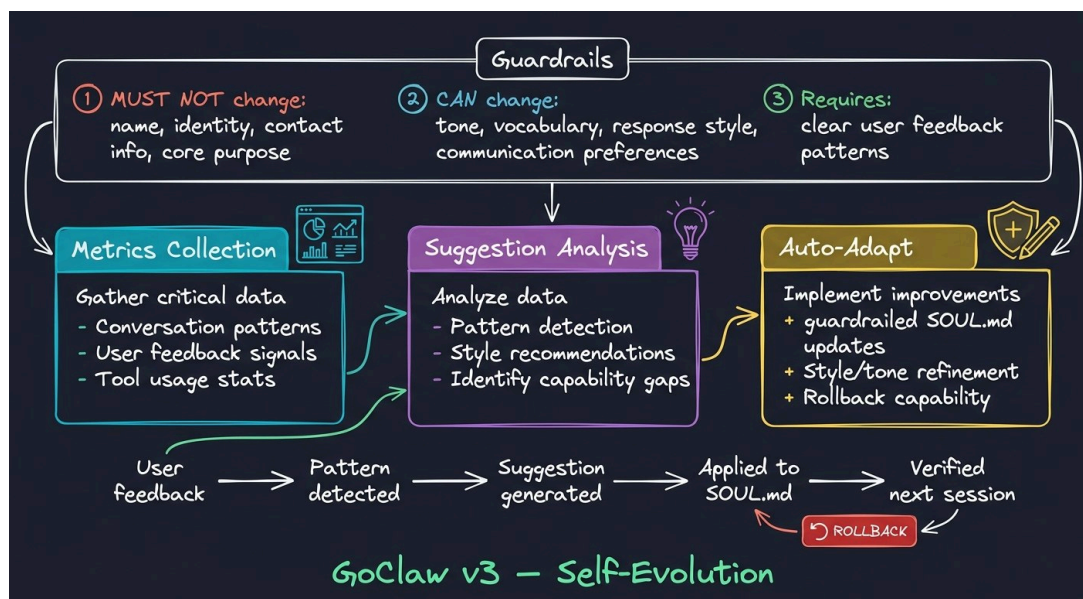
Details: [Agent Teams docs](#)

### Knowledge Vault



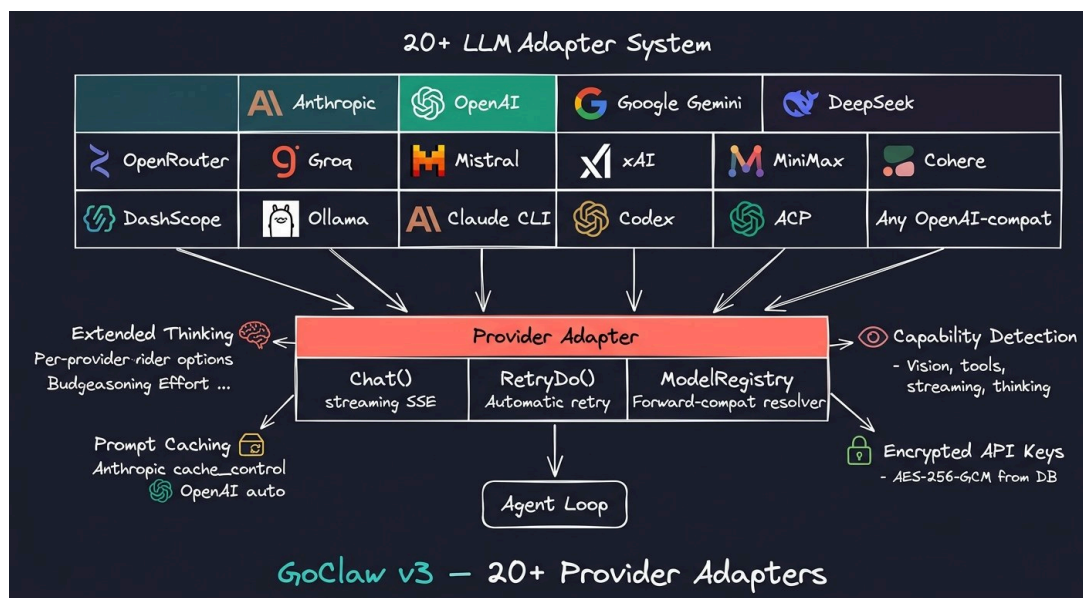
Document registry with **[[wikilinks]]** for bidirectional linking. Hybrid search combines full-text (BM25) and semantic (pgvector) for precise retrieval. Filesystem sync keeps vault in sync with on-disk files.

## Self-Evolution



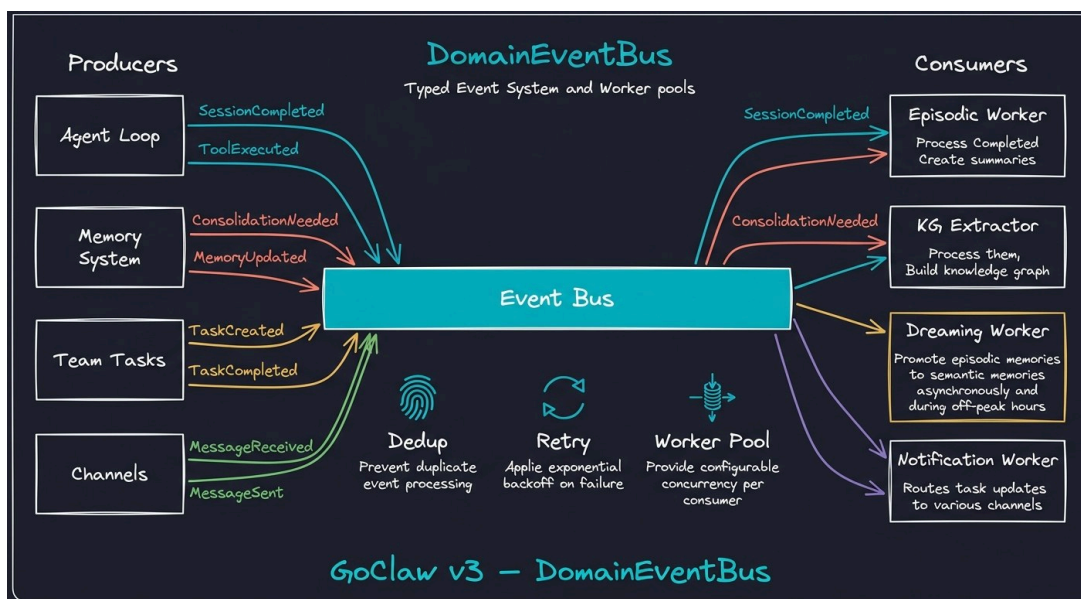
Agents improve themselves through a 3-stage guardrailed pipeline: metrics collection → suggestion analysis → auto-adaptation. Can refine communication style and domain expertise (CAPABILITIES.md) — but never change identity, name, or core purpose.

## Provider Adapters



20+ LLM providers unified through a single adapter interface. Capability-based routing, encrypted API keys (AES-256-GCM), extended thinking support per-provider, and prompt caching for Anthropic + OpenAI.

## Event-Driven Architecture



Typed domain events power the consolidation pipeline — session summaries, knowledge graph extraction, and dreaming promotion all run asynchronously via worker pools with dedup and retry.

## Built-in Tools

30+ tools across 8 categories:

| Category   | Tools  | Description   |
|------------|--|---|
| Filesystem | read_file, write_file, edit_file, list_files, search, glob | File operations with virtual FS routing                 |
| Runtime    | exec, browser  | Shell commands (approval workflow) + browser automation |
| Web        | web_search, web_fetch                                      | Search (Brave, DuckDuckGo) + content extraction         |
| Memory     | memory_search, memory_get, knowledge_graph_search          | 3-tier memory + KG traversal                            |
| Media      | create_image, create_audio, create_video, read_*, tts      | Generation + analysis (multi-provider)                  |
| Skills     | skill_search, use_skill, skill_manage                      | BM25 + semantic hybrid search                           |
| Teams      | team_tasks, spawn, delegate, message                       | Task board + orchestration + messaging                  |
| Automation | cron, heartbeat, sessions_*                                | Scheduling + session management                         |

Full tool reference at [docs.goclaw.sh](https://docs.goclaw.sh)

## Documentation

Full documentation at [docs.goclaw.sh](https://docs.goclaw.sh) — or browse the source in [goclaw-docs/](https://github.com/nextlevelbuilder/goclaw)

| Section                         | Topics  |
|---------------------------------|---|
| <a href="#">Getting Started</a> | Installation, Quick Start, Configuration, Web Dashboard Tour  |
| <a href="#">Core Concepts</a>   | Agent Loop, Sessions, Tools, Memory, Multi-Tenancy            |
| <a href="#">Agents</a>          | Creating Agents, Context Files, Personality, Sharing & Access |
| <a href="#">Providers</a>       | Anthropic, OpenAI, OpenRouter, Gemini, DeepSeek, +15 more     |
| <a href="#">Channels</a>        | Telegram, Discord, Slack, Feishu, Zalo, WhatsApp, WebSocket   |

| Section                     | Topics  |
|-----------------------------|---|
| <a href="#">Agent Teams</a> | Teams, Task Board, Messaging, Delegation & Handoff                |
| <a href="#">Advanced</a>    | Custom Tools, MCP, Skills, Cron, Sandbox, Hooks, RBAC             |
| <a href="#">Deployment</a>  | Docker Compose, Database, Security, Observability, Tailscale      |
| <a href="#">Reference</a>   | CLI Commands, REST API, WebSocket Protocol, Environment Variables |

## Testing

```
go test ./... # Unit tests
go test -v ./tests/integration/ -timeout 120s # Integration tests (requires running gateway)
```

## Project Status

See [CHANGELOG.md](#) for detailed feature status including what's been tested in production and what's still in progress.

## Acknowledgments

GoClaw was originally inspired by the [OpenClaw](#) project architecture.

## License

[CC BY-NC 4.0](#) — Creative Commons Attribution-NonCommercial 4.0 International

## Star History

### Releases 492

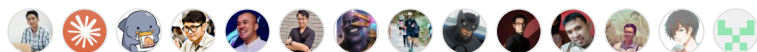
GoClaw Lite v3.9.1 Latest  
5 days ago

[+ 491 releases](#)

### Packages 2

- goclaw
- goclaw-web

### Contributors 65



[+ 51 contributors](#)

### Languages

