Taller de Alto Impacto

AGENTES DE IA

Gabriel Alzate — Ingeniero y Magíster (Univ. Distrital) con visión estratégica en transformación digital a gran escala.

- Asesor de la Presidencia de la República de Colombia en adopción tecnológica.
- Fundador de **DatosCol**, firma líder en implementación de IA y modernización de procesos.
- Especialista en arquitecturas robustas, automatización OSINT y despliegue de LLMs.

Articulando el rigor académico con la ejecución técnica avanzada para el sector público y privado.

@gabalzate



```
> ESTRATEGIA_DIGITAL
  > LLM_DEPLOYMENT
 > OSINT_AUTOMATION
```

Taller de Alto Impacto

AGENTES DE IA

En este taller aprenderemos como con Inteligencia Artificial y agentes podemos automatizar lo que hacemos en nuestros día a día y en el trabajo, como es la generación de informes, manejo de excel, creación de reportes, creación de propuestas y tableros.

Todo esto desde herramientas gratis y abiertas. De preferencia lleva tu portátil para salir del taller con un sistema agéntico personal que haga tu trabajo.

Articulando el rigor académico con la ejecución técnica avanzada para el sector público y privado.

@gabalzate



> ESTRATEGIA_DIGITAL
> LLM_DEPLOYMENT
> OSINT_AUTOMATION

FECHA:
Marzo 2026

FUENTE:
METR (Nonprofit Research)



Horizontes de Tiempo en IA: Midiendo la Frontera

Informe de Telemetría METR

MÉTRICA:
Horizonte de Completación de Tareas

ESTADO:
Confirmado

NotebookLM

AÑO: 2020
MODELO: GPT-2

LATENCIA: 24ms

TOKENS/S: 50

4 Segundos

LATENCIA: 24ms

24ms

TOKENS/S: 50



AÑO: 2026
MODELOS: Claude Opus 4.6 | GPT-5.3-Codex

CAPACIDAD: FRONTERA+

CONTEXTO: 1M+

> 10 Horas

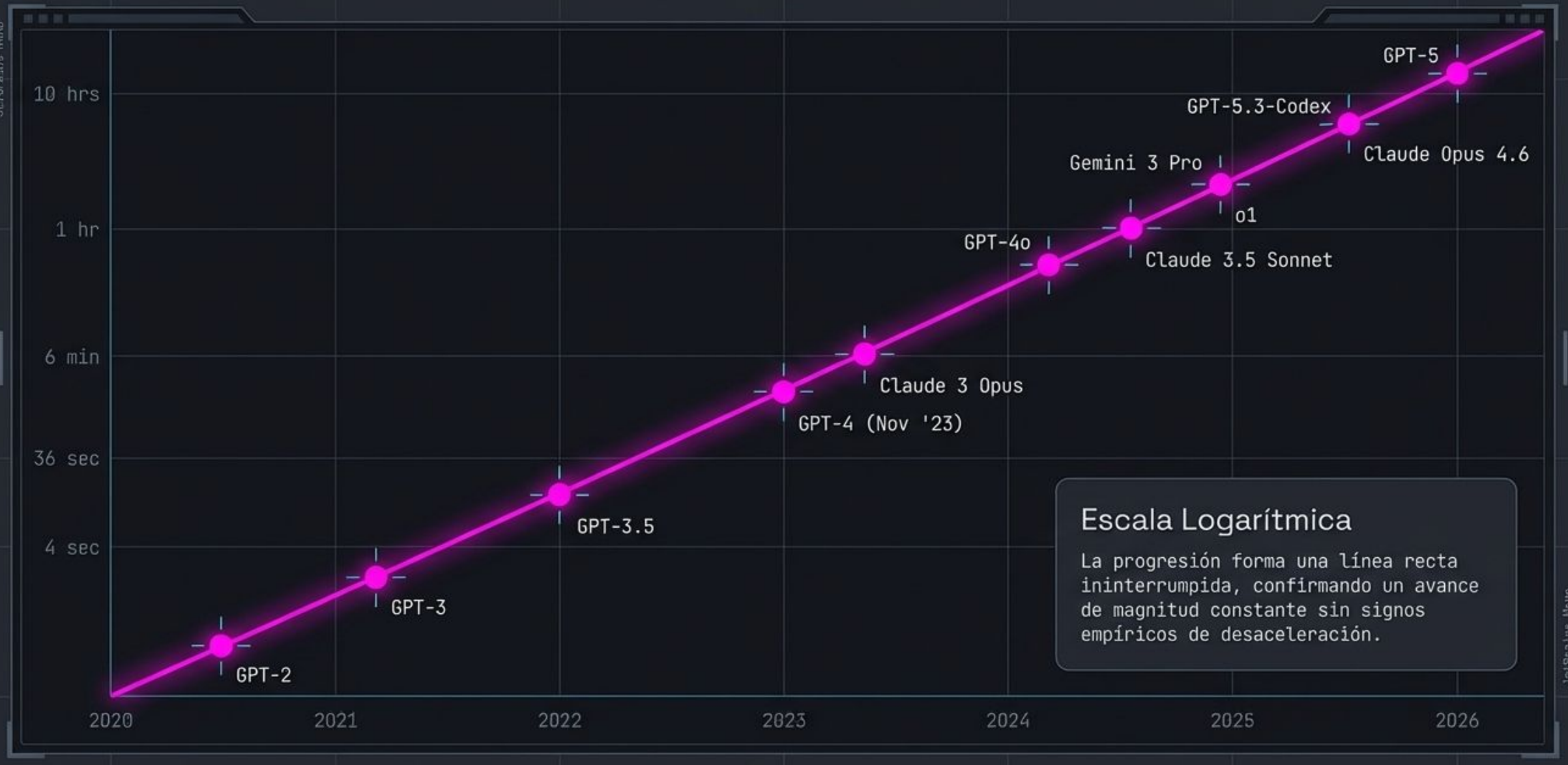
CAPACIDAD:
FRONTERA+

CONTEXTO:
1M+

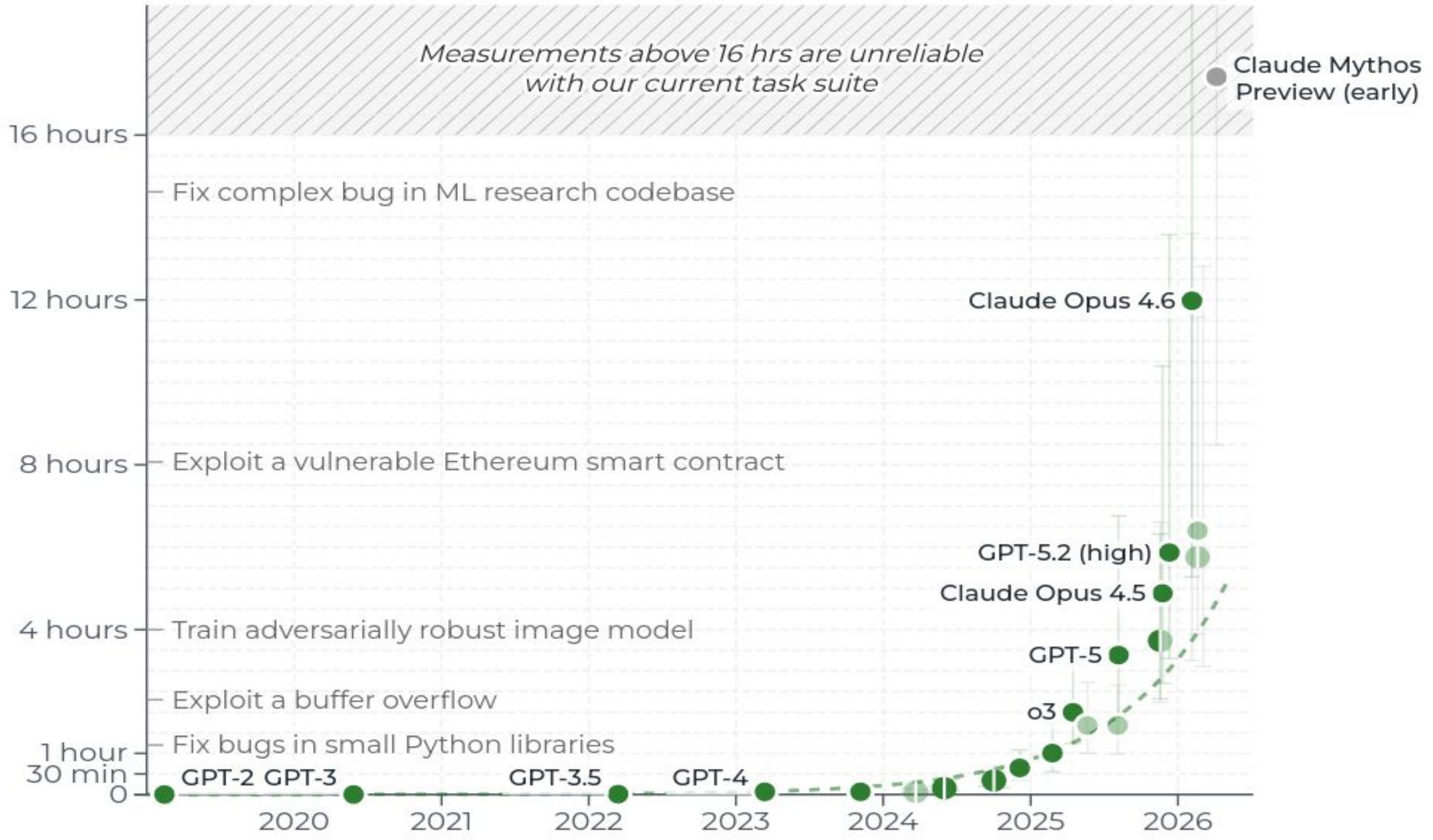
El Salto Exponencial. En solo seis años, la capacidad comprobada de la IA frontera ha escalado desde tareas triviales de segundos hasta la resolución confiable de desafíos técnicos que tomarían más de 10 horas a un profesional humano.

NotebookLM

La Evolución de la Frontera (2020-2026)



NotebookLM

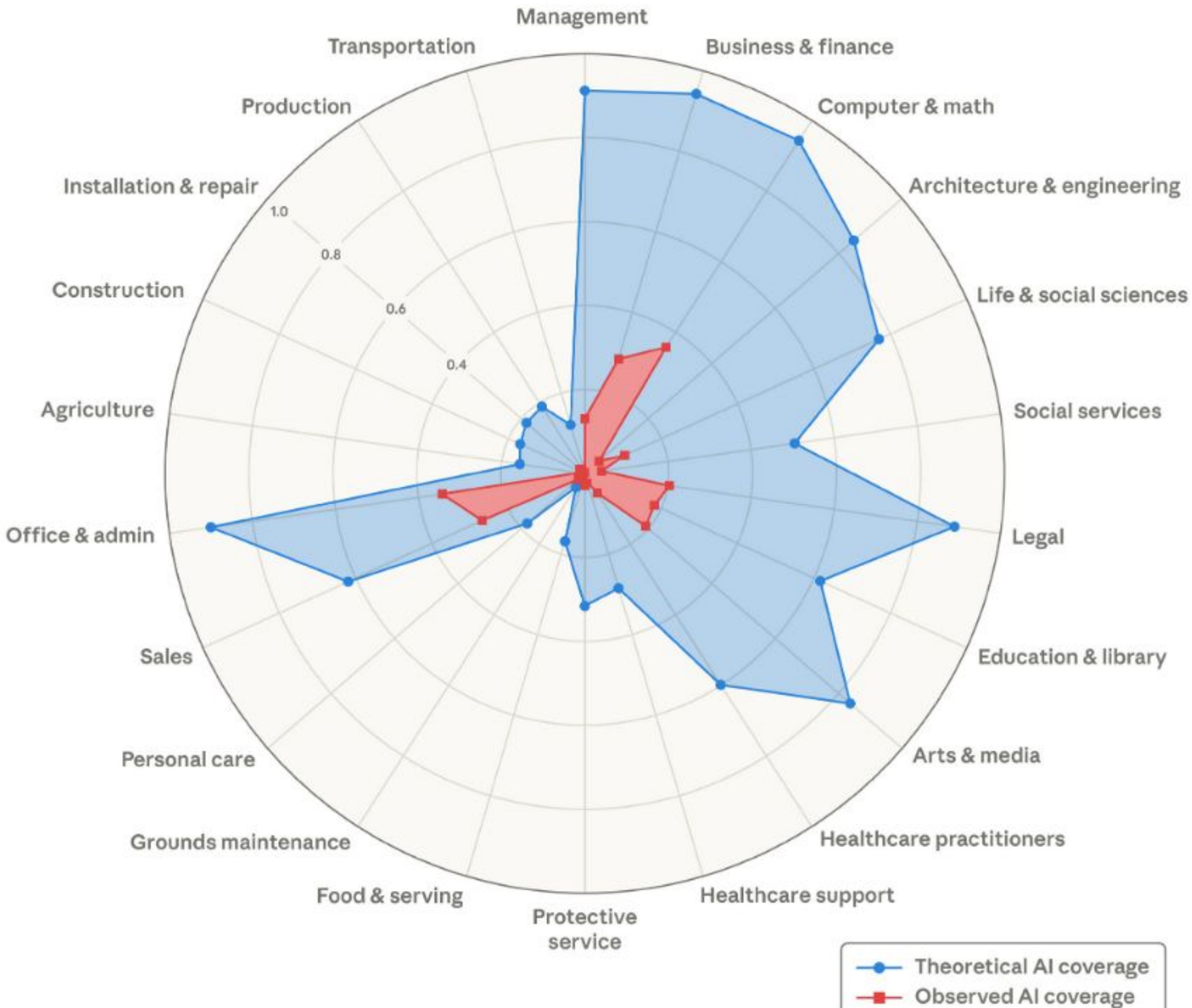


Time Horizon 1.1 (Current) Log Scale Linear Scale 50% Success 80% Success

¿Para qué se usa la IA en una empresa?

- Redacción de correos y comunicaciones
- Análisis y obtención de información
- Chat e investigación de propósito general
- Generación de código y automatización
- Detección de amenazas de ciberseguridad
- Extracción de metadatos de documentos
- Atención al cliente
- Revisión y análisis de documentos legales

Theoretical capability and observed usage by occupational category



Bienvenido a la Era del Supertrabajador

En 2026, la IA ya no es una herramienta externa, es una extensión de tu impacto individual. El 75% de los trabajadores del conocimiento ya la integran en su flujo diario para escalar su creatividad.



De Asistentes Reactivos a Agentes Autónomos

El Salto Agéntico

Dejamos atrás los chatbots que solo responden.
Los agentes actuales planifican y ejecutan tareas multietapa sin intervención constante.
La adopción de IA agéntica crecerá un 100% en los próximos dos años.



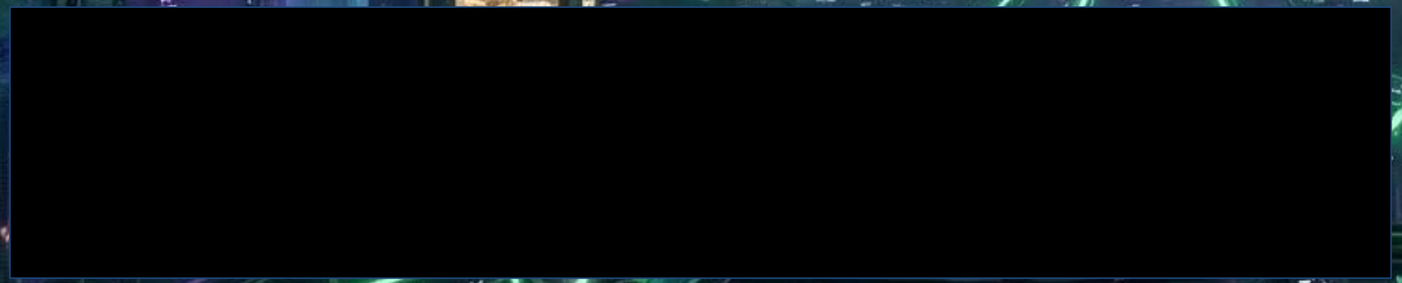
Adiós a la Amnesia Digital

Memoria Persistente (Contexto)



¿Qué es un Agente de IA?

Un **agente** es un **sistema autónomo** que no solo genera texto, sino que planifica, ejecuta y toma decisiones para alcanzar un objetivo. **El 23% de las empresas ya están escalando agénticos en 2026.**



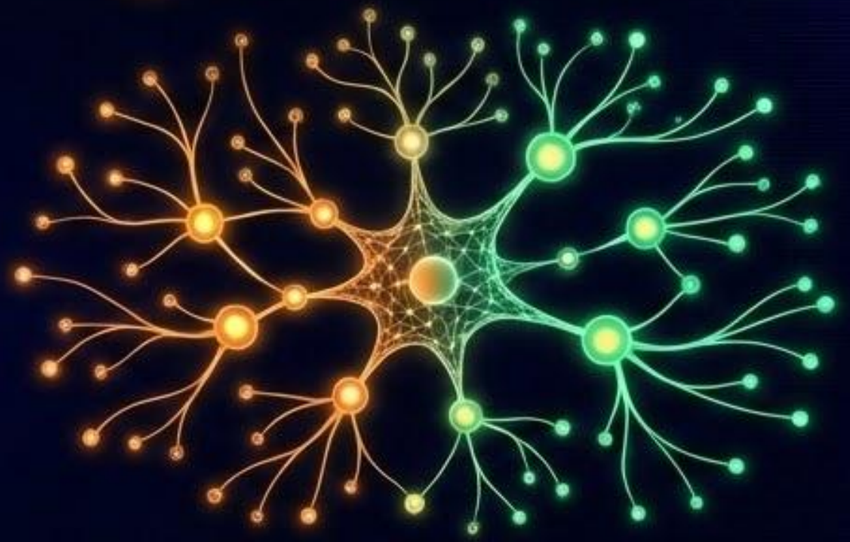
La Diferencia: Esperar vs. Actuar

Chat Reactivo



Espera tu comando.
Reactivo.

Agente Proactivo



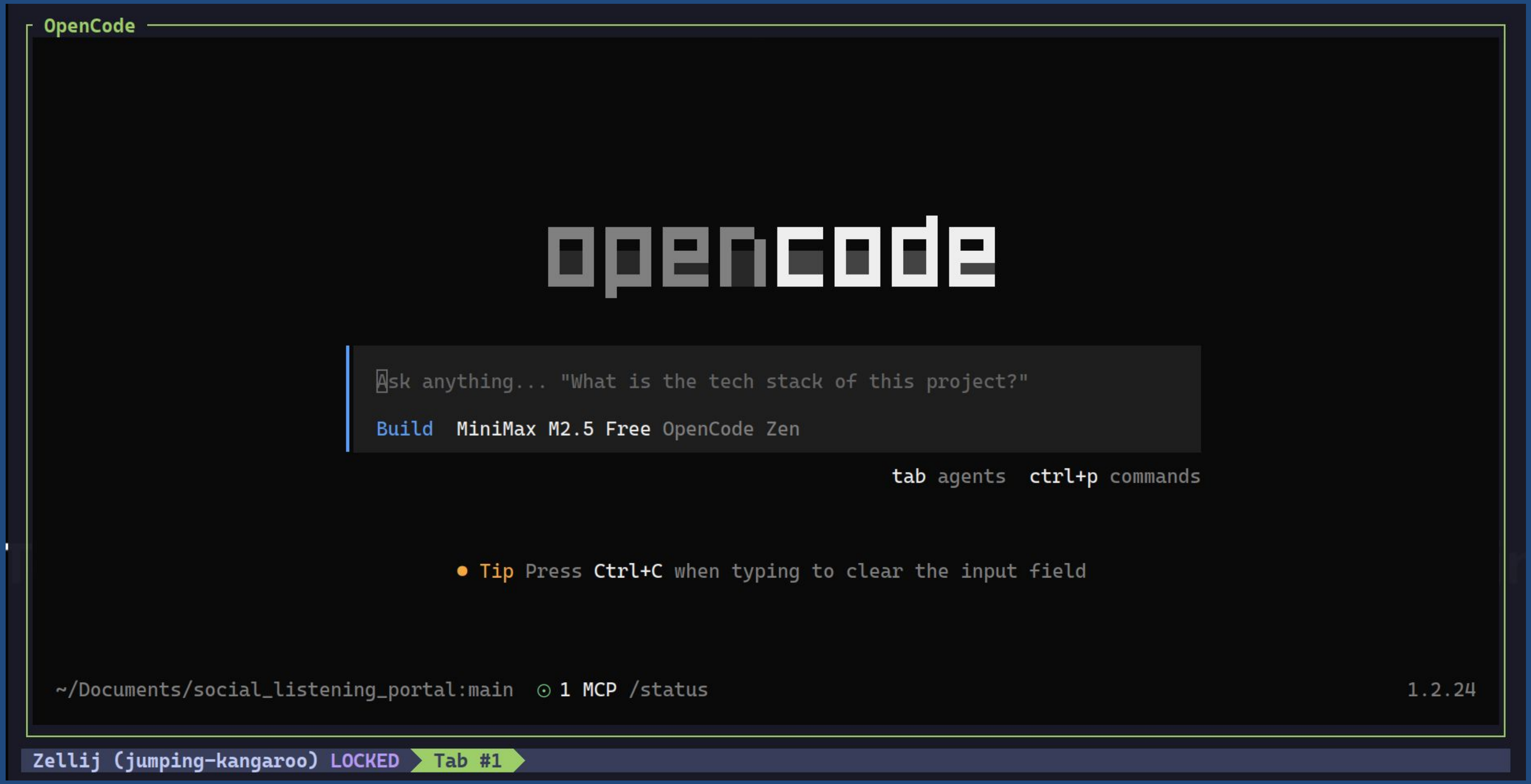
Monitorea señales y actúa solo.
Proactivo.

El cambio agéntico permite ahorrar hasta un 70% de tiempo en procesos administrativos.

El Agente "Tiene Manos"

A diferencia de un chat que solo escribe, un agente puede usar herramientas: leer archivos locales, ejecutar comandos de terminal y llamar a APIs externas de forma independiente.




OpenCode
opencode
Ask anything... "What is the tech stack of this project?"
Build MiniMax M2.5 Free OpenCode Zen
tab agents ctrl+p commands
Tip Press Ctrl+C when typing to clear the input field
~/Documents/social_listening_portal:main ⊙ 1 MCP /status 1.2.24
Zellij (jumping-kangaroo) LOCKED Tab #1

LLM Leaderboard

This Week

Compare the most popular models on OpenRouter

1.	**Hy3 preview (free)** by tencent	3.75T tokens ↑74%	6.	**DeepSeek V4 Flash** by deepseek	975B tokens ↑58%
2.	**Kimi K2.6** by moonshotai	1.72T tokens ↓9%	7.	**DeepSeek V3.2** by deepseek	836B tokens ↓15%
3.	**Claude Sonnet 4.6** by anthropic	1.43T tokens ↑5%	8.	**MiniMax M2.7** by minimax	766B tokens ↑8%
4.	**Claude Opus 4.7** by anthropic	1.14T tokens ↑21%	9.	**DeepSeek V4 Pro** by deepseek	763B tokens ↑187%
5.	**Gemini 3 Flash Preview** by google	1.06T tokens ↑8%	10.	**Grok 4.1 Fast** by x-ai	730B tokens ↑13%

Show more

MCP (Model Context Protocol): La Red Micelial del Conocimiento Digital

El Ecosistema Compartido: El MCP es un estándar abierto diseñado para unificar cómo las inteligencias artificiales (LLMs) se nutren e interactúan con fuentes de datos externas y herramientas locales.

La Conexión Simbiótica: Actúa como un 'traductor universal' o una red micelial subyacente. En lugar de forzar a los desarrolladores a construir conexiones individuales y extractivas para cada herramienta, el MCP establece una única raíz de comunicación estandarizada, fluida y segura.

Cosecha de Beneficios:

- **Cultivo Sostenible:** Simplifica el esfuerzo de los creadores de IA, evitando la fragmentación del código y promoviendo un ecosistema de desarrollo más orgánico.
- **Agentes Vivos:** Permite que las IA florezcan con la capacidad de tomar acciones y absorber información fresca en tiempo real de su entorno.
- **Equilibrio y Seguridad:** Garantiza que los datos se compartan de manera interoperable, transparente y protegida, manteniendo la armonía digital.

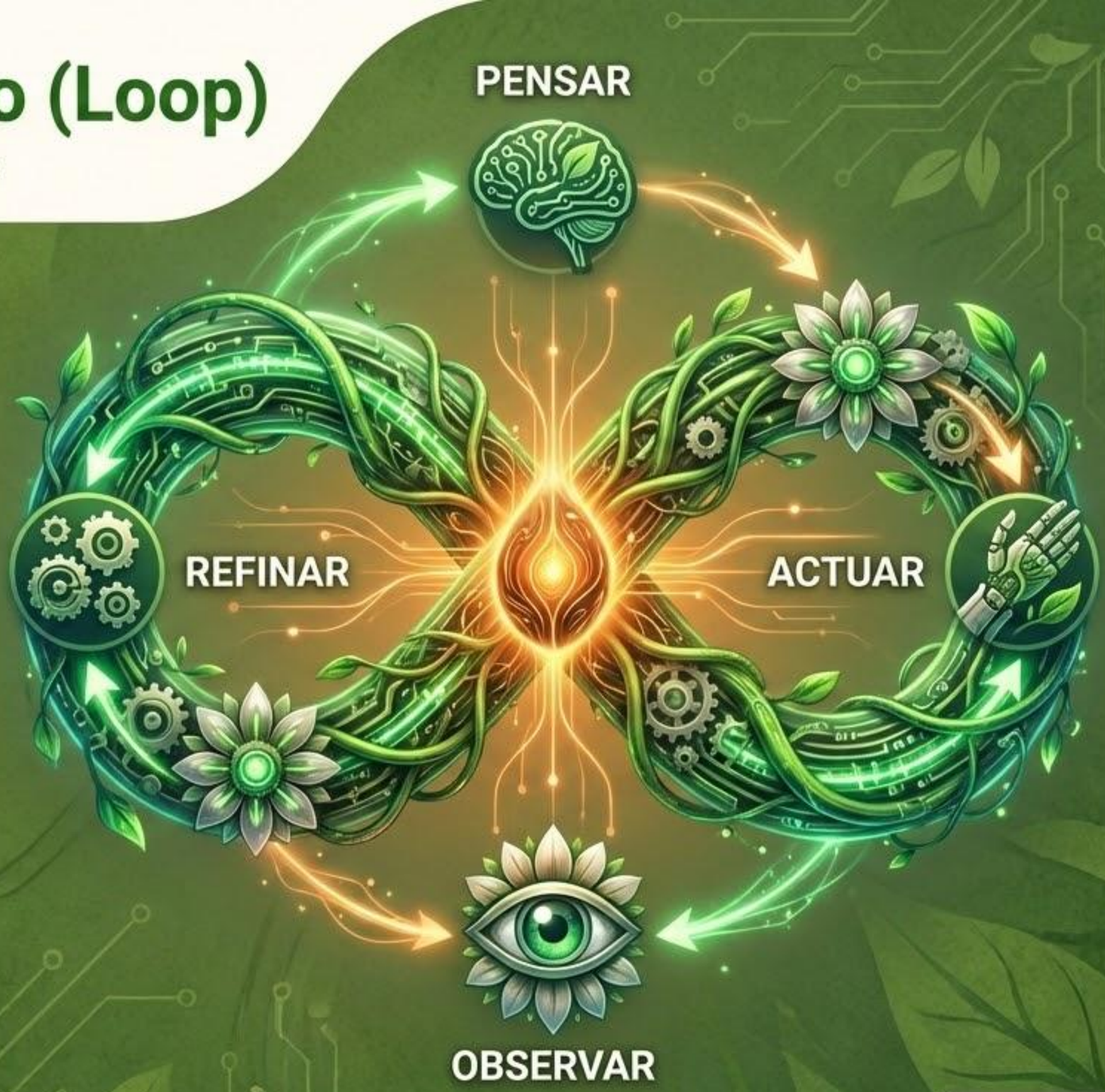


El Ciclo de Pensamiento (Loop)

Pensar → Actuar → Observar → Refinar

Los agentes operan en un bucle continuo. Evalúan el resultado de sus acciones y corrigen su plan sobre la marcha sin que tengas que pedírselo.

Esto **reduce errores** en un **60%**



Gabriel Alzate

Skills: Expertos bajo Demanda

Los agentes cargan “Skills” (paquetes de instrucciones específicos) solo cuando son necesarios.
Esto evita la saturación de memoria y mejora la precisión lógica en un 40%.



Gabriel Alzate

What is a skill?

A skill is a folder containing:

- **SKILL.md** (required): Instructions in Markdown with YAML frontmatter
- **scripts/** (optional): Executable code (Python, Bash, etc.)
- **references/** (optional): Documentation loaded as needed
- **assets/** (optional): Templates, fonts, icons used in output

Core design principles

Progressive Disclosure

Skills use a three-level system:

- **First level (YAML frontmatter):** Always loaded in Claude's system prompt. Provides just enough information for Claude to know when each skill should be used without loading all of it into context.
- **Second level (SKILL.md body):** Loaded when Claude thinks the skill is relevant to the current task. Contains the full instructions and guidance.
- **Third level (Linked files):** Additional files bundled within the skill directory that Claude can choose to navigate and discover only as needed.

This progressive disclosure minimizes token usage while maintaining specialized expertise.

Technical requirements

File structure

```
your-skill-name/
├── SKILL.md              # Required - main skill file
├── scripts/              # Optional - executable code
│   ├── process_data.py # Example
│   └── validate.sh # Example
├── references/           # Optional - documentation
│   ├── api-guide.md # Example
│   └── examples/ # Example
└── assets/               # Optional - templates, etc.
    └── report-template.md # Example
```

Critical rules

SKILL.md naming:

- Must be exactly `SKILL.md` (case-sensitive)
- No variations accepted (SKILL.MD, skill.md, etc.)

Skill folder naming:

- Use kebab-case: `notion-project-setup` ✅
- No spaces: `Notion Project Setup` ❌
- No underscores: `notion_project_setup` ❌
- No capitals: `NotionProjectSetup` ❌

No README.md:

- Don't include README.md inside your skill folder
- All documentation goes in SKILL.md or references/
- Note: when distributing via GitHub, you'll still want a repo-level README for human users — see Distribution and Sharing.

YAML frontmatter: The most important part

The YAML frontmatter is how Claude decides whether to load your skill. Get this right.

Minimal required format

```
---
name: your-skill-name
description: What it does. Use when user asks to [specific phrases].
---
```

That's all you need to start.

Field requirements

name (required):

- kebab-case only
- No spaces or capitals
- Should match folder name

description (required):

- **MUST include BOTH:**
  - What the skill does
  - When to use it (trigger conditions)
- Under 1024 characters
- No XML tags (< or >)
- Include specific tasks users might say
- Mention file types if relevant

Criterios de Selección: El Semáforo Agéntico



Las Reglas de la Poda Digital

Reglas de Oro para tu “Kill List”

Para que la poda sea efectiva, aplica estas 3 leyes:

1. **Simplifica antes de Automatizar:** Si el proceso manual está roto, la IA solo lo hará fallar más rápido.
2. **Regla 80/20:** Prioriza las tareas que consumen el 80% de tu tiempo administrativo para liberar al menos 5-10 horas semanales.
3. **Contrato de Entrada/Salida:** Toda tarea en la lista debe tener un input claro (ej. email) y un output verificable (ej. cita en calendario).



Acción: Diseña tu Ruta de Automatización

No intentes programar; usa a OpenCode como tu arquitecto consultor.



El Mapa del Supertrabajador: De la Tarea al Sistema

Deja de pedir "respuestas" y empieza a diseñar "procesos". Un flujo agéntico bien mapeado reduce las alucinaciones de la IA en un 60% al descomponer metas grandes en micro-pasos lógicos. Define tu contrato de trabajo:

Input (Entrada): Los datos brutos (ej. un hilo de 50 emails).

Proceso: Los pasos de razonamiento (ej. clasificar por urgencia).

Output (Salida): El entregable final (ej. un resumen de 3 puntos en Slack).



Obsidian CLI

Obsidian CLI is a command line interface that lets you control Obsidian from your terminal for scripting, automation, and integration with external tools.

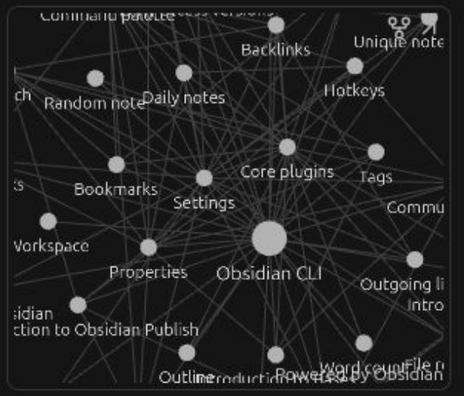
Anything you can do in Obsidian you can do from the command line. Obsidian CLI even includes developer commands to access developer tools, inspect elements, take screenshots, reload plugins, and more.



⚠ Requires Obsidian 1.12 installer

Using the CLI requires the Obsidian 1.12 installer. See the installer version update guide.

INTERACTIVE GRAPH



ON THIS PAGE

- Install Obsidian CLI
- Get started
  - Run a command
  - Use the terminal interface
- Examples
  - Everyday use
  - For developers
- How to
  - Use parameters and flags
  - Target a vault
  - Target a file
  - Copy output
- General commands
  - help
  - version
  - reload
  - restart
- Bases
  - bases
  - base:views
  - base:create

Playwright MCP

A Model Context Protocol (MCP) server that provides browser automation capabilities using Playwright. This server enables LLMs to interact with web pages through structured accessibility snapshots, bypassing the need for screenshots or visually-tuned models.

Playwright MCP vs Playwright CLI

This package provides MCP interface into Playwright. If you are using a **coding agent**, you might benefit from using the CLI+SKILLS instead.

- **CLI**: Modern **coding agents** increasingly favor CLI–based workflows exposed as SKILLs over MCP because CLI invocations are more token-efficient: they avoid loading large tool schemas and verbose accessibility trees into the model context, allowing agents to act through concise, purpose-built commands. This makes CLI + SKILLs better suited for high-throughput coding agents that must balance browser automation with large codebases, tests, and reasoning within limited context windows.
  Learn more about Playwright CLI with SKILLS.

Deployments 53

allow-publishing-docker-to-acr
last month

allow-mcp-registry-publishing

+ 51 deployments

Contributors 62



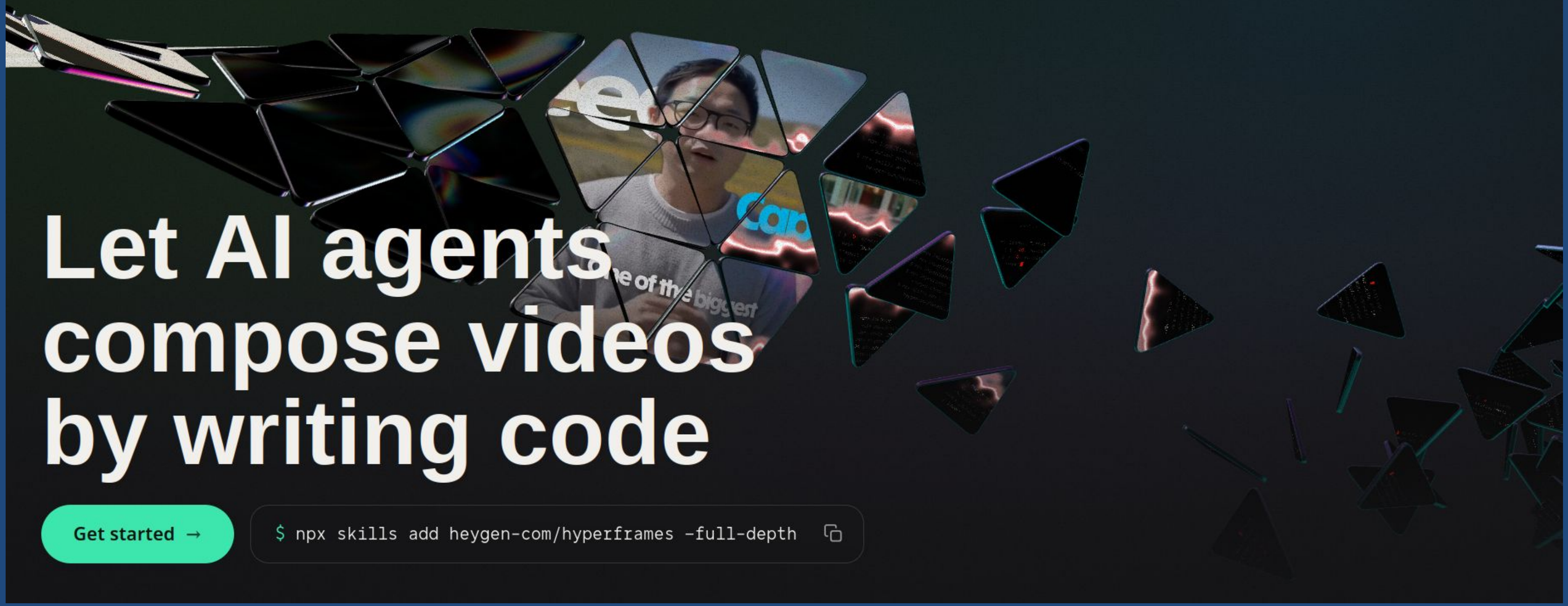
+ 48 contributors

Languages

TypeScript 54.4% CSS 34.1%
JavaScript 8.8% Dockerfile 1.7%
HTML 1.0%

HeyGen HyperFrames

Docs GitHub Catalog Playground

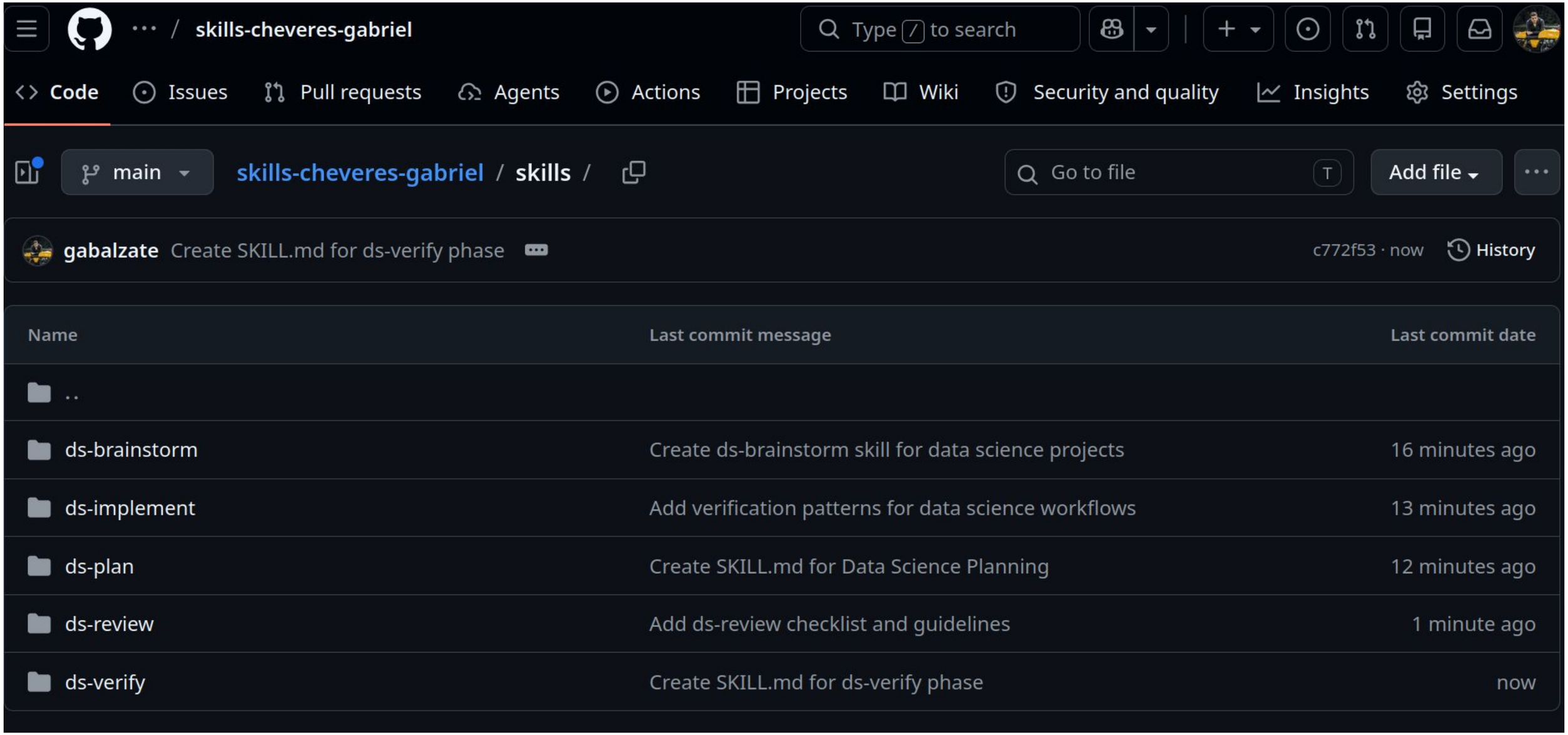


Let AI agents compose videos by writing code

Get started →

```
$ npx skills add heygen-com/hyperframes -full-depth
```

https://github.com/gabalzate/skills-cheveres-gabriel/tree/main/skills

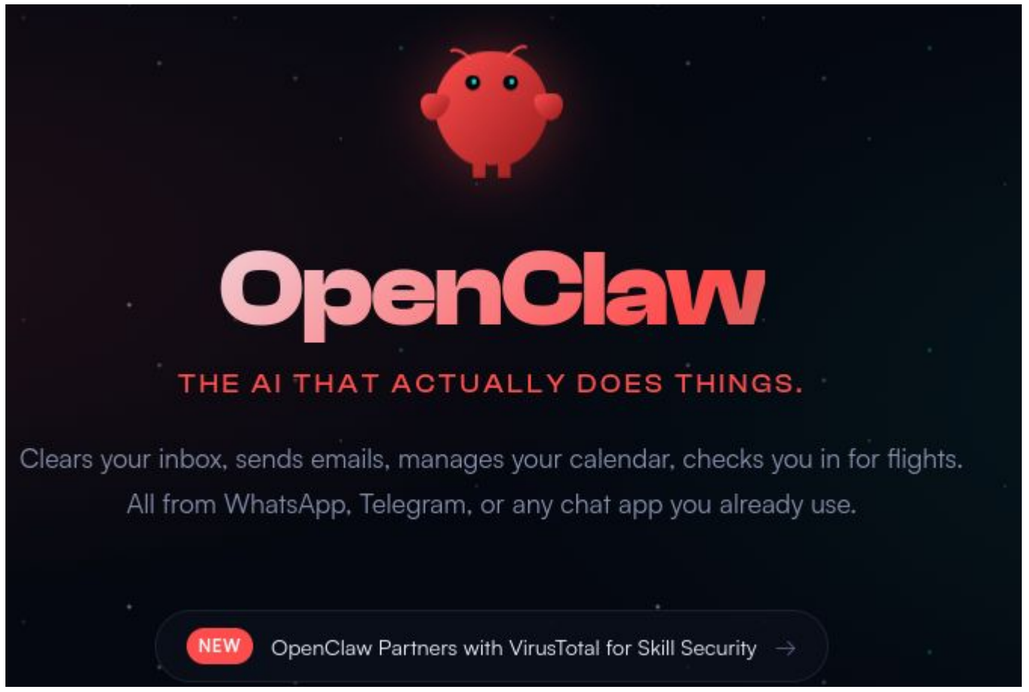


MULTI-AGENTES (ENJAMBRES)

DE UN ASISTENTE A UN EQUIPO DIGITAL

Puedes orquestar múltiples agentes en paralelo. Mientras uno investiga, otro redacta y un tercero valida. Esto acelera la productividad individual por un factor de 5x.




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.
NEW
OpenClaw Partners with VirusTotal for Skill Security


INICIO
WORKSPACE
ARQUITECTURA
NOSOTROS
HIVE
V2.0
PROYECTOS
BRIDGE
TOOLS
CONTACTO
EN
ESTADO DEL SISTEMA: ÓPTIMO
V0.0.14
REAL-TIME SYNC
Una colmena de agentes.
Completamente tuya.
Un gateway local para todos tus canales. Sin cloud. Sin suscripciones. Sin límites.
HIVE COLLECTIVE V2.0
SECURE LATTICE
Instalar ahora
VER EN GITHUB


NOUS RESEARCH PRESENTS
HERMES 4
FEATURING NOUS CHAT REVAMPED

OPEN SOURCE • MIT LICENSE

AN AGENT THAT GROWS WITH YOU.

Not a coding copilot tethered to an IDE or a chatbot wrapper around a single API. An autonomous agent that lives on your server, remembers what it learns, and gets more capable the longer it runs.

1. INSTALL COPY

```
curl -fssl https://raw.githubusercontent.com/nousresearch/hermes-agent/main/scripts/install.sh | bash
```

2. CONFIGURE COPY

```
hermes setup
```

FEATURES

LIVES WHERE YOU DO

Telegram, Discord, Slack, WhatsApp, Signal, Email, CLI — and a growing list of platforms. Start on one, pick up on another.

GROWS THE LONGER IT RUNS

Persistent memory and auto-generated skills — it learns your projects and never forgets how it solved a problem.

SCHEDULED AUTOMATIONS

Natural language cron scheduling for reports, backups, and briefings — running unattended through the gateway.

DELEGATES & PARALLELIZES

Isolated subagents with their own conversations, terminals, and Python RPC scripts for zero-context-cost pipelines.

REAL SANDBOXING

Five backends — local, Docker, SSH, Singularity, Modal — with container hardening and namespace isolation.

FULL WEB & BROWSER CONTROL

Web search, browser automation, vision, image generation, text-to-speech, and multi-model reasoning.

LESS ^

TOOLS	40+ built-in — web search, terminal, file system, browser automation, vision, image generation, text-to-speech, code execution, subagent delegation, memory, task planning, cron scheduling, multi-model reasoning, and more.
PLATFORMS	Telegram, Discord, Slack, WhatsApp, Signal, Email, CLI — and a growing list of platforms, all from a single gateway. Connect to Nous Portal, OpenRouter, or any OpenAI-compatible API.
ENVIRONMENTS	Run locally, in Docker, over SSH, on Modal, Daytona, or Singularity. Container hardening with read-only root, dropped capabilities, and namespace isolation.
SKILLS	40+ bundled skills covering MLOps, GitHub workflows, research, and more. The agent creates new skills on the fly and shares them via the open agentskills.io format.
RESEARCH	Batch trajectory generation with parallel workers and checkpointing. Atropos integration for RL training. Export to ShareGPT for fine-tuning with trajectory compression.