CASE STUDY

# From Chatbot to Co-Worker

The OpenClaw story — how a personal side project became the fastest-growing open-source project in history, and what it tells us about the age of AI agents.

June 2026 • Adventures in AI • 12 min read

Download as PDF

## Part 1: The Origin

### From PDFs to Agents

Peter Steinberger isn't a newcomer. He founded PSPDFKit in 2011, building it into one of the most successful PDF SDK companies in the world — used by enterprises like Airbus, Disney, and SAP. He ran it for over a decade before exiting, wealthy enough that he didn't need to build anything ever again.

But in early 2025, Steinberger started experimenting with AI coding agents. His breakthrough wasn't technical — it was philosophical. He realised that the AI tools most people were using (ChatGPT, Claude, Gemini) were fundamentally limited by their design: they were chatbots. You asked a question, you got an answer. The interaction started when you typed and ended when the response finished.

Steinberger's insight was simple but radical: **what if the AI didn't wait for you to ask?**

### The First Release: November 2025

On 24 November 2025, Steinberger released the first public version of what would become OpenClaw. It started under the name *Clawdbot* — a playful pun on "Claude" (the Anthropic model it primarily used) combined with the lobster mascot that became the project's identity.

The core idea: instead of a chat interface, the agent lived in your messaging apps — WhatsApp, Telegram, iMessage, Discord, Slack. You could talk to it the same way you'd talk to a colleague. But unlike a chatbot, it could *do things*: read files on your computer, browse the web, manage your calendar, send emails, execute code. It ran locally on your machine, with your permissions, using your accounts.

> ***"People don't want smarter chatbots. They want co-workers. The difference isn't incremental — it's categorical."***

## Part 2: The Explosion

## Going Viral

What happened next was extraordinary even by the standards of the AI hype cycle. Within weeks of its public release, Clawdbot's GitHub repository was gaining stars at a rate that surpassed any open-source project in history.

To put that in perspective: **Linux** took over 20 years to reach 218,000 stars. **React** took 8 years to reach 247,000 stars. **OpenClaw** reached 250,000 stars in approximately 10 weeks.

By March 2026, it had briefly surpassed React to become one of the most-starred repositories on all of GitHub. The velocity was unprecedented — tens of thousands of stars per day at its peak.

### Why It Caught Fire

Several factors converged:

**It solved a real problem.** People were tired of chatbots that could only talk. OpenClaw could actually *act*.

**It was local-first.** In an era of data breaches and privacy scandals, running your AI agent on your own machine resonated deeply.

**It was model-agnostic.** It could use Claude, GPT, Gemini, local models via Ollama, or any of 20+ other providers.

**The timing was right.** By late 2025, millions had used ChatGPT enough to understand what LLMs could do — and hit the wall of what chat-only interfaces *couldn't* do.

**The developer experience was excellent.** Within minutes of cloning the repo, you could have an agent running on your Mac.

### The Naming Saga

What followed was one of the fastest rebrands in technology history:

**January 27, 2026:** Renamed to **Moltbot** (lobsters molt when they grow — the crustacean theme continued)

**January 29–30, 2026:** Renamed again to **OpenClaw** (emphasising open-source values and model-agnostic independence)

Three brand names in roughly one week. The community found it amusing rather than confusing — the lobster mascot, affectionately named "Clawrence," became a rallying point.

## Part 3: The Crisis

### ClawHavoc

In February 2026, just as OpenClaw was hitting its stride, disaster struck.

Security researchers at Koi Security disclosed **ClawHavoc** — a coordinated supply-chain attack on ClawHub, OpenClaw's community skill marketplace. The numbers were stark:

**341 malicious skills** identified in an initial audit of ~2,857 total skills (~12% of the registry)

Later analyses raised the count to **824–1,184 malicious entries**

Attack methods included **Atomic Stealer (AMOS)** on macOS, keyloggers, reverse shells, and crypto wallet theft

Delivery mechanism: social engineering — skills posing as useful tools with fake "prerequisite" instructions

Around the same time, **CVE-2026-25253** was disclosed — a one-click remote code execution vulnerability via token leakage. The headlines were brutal.

### The Response

The OpenClaw Foundation responded with a multi-layered security overhaul:

- **VirusTotal integration** — every skill uploaded to ClawHub now gets scanned
- **Verified publisher programme** — trusted authors get a verification badge
- **Skill Workshop governance** (May 2026) — a formal proposal, review, quarantine, and rollback system
- **Externalised plugin architecture** — core runtime components moved to npm packages with proper supply-chain security

> *"ClawHavoc was a genuine crisis. But the open-source response was faster and more thorough than any closed-source incident response could have been."*

## Part 4: The Foundation

### Steinberger Joins OpenAI

On February 15, 2026, Peter Steinberger announced he was joining OpenAI to work on next-generation personal agents. The immediate fear: would OpenAI swallow OpenClaw?

### Independent by Design

The OpenClaw Foundation was established as an **independent non-profit**, explicitly modelled on the Ghostty Foundation structure:

- **Community-elected board** of maintainers — not appointed by any company

- **MIT licence** — fully permissive, ensuring the code stays free forever
- **OpenAI as minority sponsor** — financial support only, no governance privileges or roadmap influence
- **Public RFC process** — all major decisions go through open discussion and maintainer votes

Sam Altman confirmed: *"OpenClaw will live in a foundation as an open source project that OpenAI will continue to support."*

## Part 5: The Ecosystem

By mid-2026, OpenClaw describes itself as a "federation of ~70 open-source projects" sharing a runtime and protocols:

- **GitHub Stars:** 300K+ (briefly #1 most-starred)
- **ClawHub Skills:** 13,000+ community contributions
- **Messaging Integrations:** 50+ (WhatsApp, Telegram, Slack, Discord, Teams, iMessage, Signal, etc.)
- **AI Provider Plugins:** 20+ (OpenAI, Anthropic, Google, Ollama, OpenRouter, etc.)
- **Community Contributors:** 400+

### Enterprise Adoption

- **NVIDIA** (April 2026): Published on long-running autonomous agents and organisational impact
- **Red Hat** (March 2026): Detailed guide on operationalising OpenClaw on OpenShift
- **AWS Marketplace:** One-click AMI for production browser automation

# Part 6: What This Means — 7 Lessons

### 1 Agents Beat Chatbots

The explosive growth wasn't driven by better language models. It was driven by giving those models *hands*.

### 2 Local-First Has Legs

In a world where every AI company wants to own your data, running locally struck a nerve. Your data on your hardware isn't a niche feature.

### 3 Model-Agnostic Is Model-Savvy

Tying your platform to a single AI provider creates dependency risk. OpenClaw's flexibility to switch models is a strategic advantage, not just a convenience.

### 4 Open Source Survives Crises Better

When 341 malicious skills were discovered, the entire community could inspect, verify, and fix. You can't do that with a black box.

### 5 Foundation Governance Can Work

Community-elected maintainers, public RFC processes, sponsor money without sponsor control — a template for other AI infrastructure projects.

### 6 Security Is the Agent Problem

Autonomy requires access, and access creates risk. Every agent platform, open or closed, must solve this fundamental challenge.

7 **Speed of Adoption Is a Double-Edged Sword**

Zero to 300K stars in months is extraordinary. It also means millions of users installing software before governance structures have matured.

## Part 7: Our Experience

Full disclosure: we use OpenClaw daily. This case study isn't just a historical exercise — it's informed by months of hands-on experience running autonomous AI agents for real business tasks.

Our setup runs 24/7 on a Mac mini, orchestrating a team of specialised agents: research (Gav), development (Q), UI design (Mel), adversarial review (Kieran), and coordination (Bee). They handle email triage, research synthesis, project management, code review, weekly strategy reports, and dozens of other tasks that would otherwise consume hours of human attention.

What we've learned:

**The always-on model changes everything.** An agent that's available 24/7, that can act proactively, is categorically different from a chatbot you visit when you remember.

**Trust is earned in small increments.** You don't start by giving an agent the keys to everything. You start with read-only tasks, verify the output, and gradually expand access.

**Security hygiene matters.** We use verified skills only, limit privileges per agent, and audit regularly.

**Multi-model is a real advantage.** Different tasks benefit from different models. Cost management depends on this flexibility.

**The human stays in the loop.** For anything external, public, or financially consequential, we require explicit approval. The agent proposes; the human disposes.

> *"The agent proposes; the human disposes."*

TIMELINE

## Key Dates

OUR TAKE

# Living With AI Agents — Every Day

We run an autonomous AI team 24/7 on a Mac mini. They handle research, development, design, code review, and dozens of other tasks. This case study isn't just history — it's informed by real, daily experience with the technology we're writing about.

Start a Conversation →

Download PDF