Omega GTM: Distribution Playbook

Distribution is everything.

The Principle

Nobody wakes up wanting "agent infrastructure." They wake up with a broken workflow, a failing agent, a context window that's full.

Meet them where they are. Solve the problem they have. Upgrade them to Omega without them noticing.

The best distribution is invisible. One command. Immediate value. Then they're in the ecosystem.

Distribution Strategy

Three flagship forks prove the full Continuity stack works everywhere. Then expand with adapters.

```bash
npx omega-code  # OpenCode fork
npx omega-claw  # OpenClaw fork
npx omega-sdk   # OpenAI Agent SDK fork
```

Three flagships. Three launches. Three viral moments. Sustained attention over weeks 1-4.

Tier 1: Flagship Forks (Weeks 1-4)

The flagships aren't adapters—they're full forks rebuilt on the complete Continuity stack.

Flagship	Fork Of	Stars	What It Proves
Omega Code	OpenCode	30K	CLI coding agents work
Omega Claw	OpenClaw	100K+	Scales to viral projects
Omega SDK	OpenAI Agent SDK	—	Framework SDKs work

Each flagship gets the complete system:

- **Persistent Memory (CAS)** — Codebase understanding compounds
- **Obligation Tracking** — Every resource managed, every task resolved
- **Attested Builds** — Cryptographic receipts for every action
- **Zeitschrift Integration** — Scope graphs catch broken references
- **Verified Correctness** — Built on Lean4 proofs

Zero-Friction Entry

True one-line installs:

```bash
npx omega-code    # instead of opencode
npx omega-claw    # instead of openclaw
npx omega-sdk     # OpenAI Agent SDK + full stack
pipx run omega-aider  # aider + memory
```

Wrap existing installs:

```bash
# One-time setup
curl -fsSL https://omega.fleek.sh/install.sh | sh

# Then prefix any command
omega opencode
omega aider
```

The alias trick (zero behavior change):

```bash
# Run once: creates omega-powered aliases
curl -fsSL https://omega.fleek.sh/alias.sh | sh

# Now normal commands have memory
opencode  # → omega-code
aider     # → omega-aider
```

Tier 2: Coding Agent Adapters (Weeks 3-6)

After flagships ship, expand with adapters for the rest:

Tool	Stars	Strategy
Cline	65K	Adapter — massive VS Code reach
OpenHands	65K	Adapter — multi-tool, autonomous
Aider	30K	Adapter — git-native, loyal community
Goose	15K	Adapter — Block/Square backing
Continue	20K	Memory backend for IDE

```bash
npx omega-cline   # Cline + memory
npx omega-hands   # OpenHands + memory
```

The pitch: "Your coding agent, but it remembers your codebase across sessions."

Tier 3: Framework Integrations (Weeks 5-8)

Framework	Stars	Strategy
LangChain	100K+	Official integration
LangGraph	10K+	Native memory layer
n8n	60K+	Community node
CrewAI	25K+	Drop-in memory
AutoGen	40K+	SDK integration

LangChain Integration:

```python
```

```
# Before: vanilla LangChain
from langchain import Agent

# After: one import change
from omega.langchain import Agent  # drop-in replacement
```

n8n Integration:

```
bash

n8n community install @omega/n8n-memory
```

Tier 4: Developer Tools (Weeks 7-12)

VS Code Extension — "Omega: Persistent AI Memory"

- Works with Copilot, Continue, Cursor, any AI assistant
- Persists context across sessions
- Shows memory usage in status bar

GitHub Action:

```
yaml

- uses: omega/setup@v1
- run: omega test
- run: omega benchmark
```

Templates:

- omega-support-bot — Customer support with infinite memory
- omega-research-agent — Research that compounds
- omega-code-reviewer — Code review that learns your codebase

---

Growth Hacks

Hack 1: The Benchmark Play

```
bash

npx omega-bench my_agent.py
```

```
Tool calling success: 47% → 89% (with Omega)
Context utilization: 23% → 94% (with Omega)
Task completion: 34% → 87% (with Omega)

Share results: https://omega.fleek.sh/bench/abc123
```

Hack 2: The Migration Analyzer

```
bash

npx omega-analyze my_project/
```

```
Found: 3 LangChain agents, 2 raw OpenAI calls, 1 n8n workflow

Estimated improvements with Omega:
- 12 context window overflows/day → 0
- ~$847/month in redundant API calls saved

Run `npx omega-upgrade` to migrate.
```

Hack 3: The "Day 100" Content Series

Weekly content showing agents running on Omega:

- "Day 7: Agent learned the codebase structure"
- "Day 30: Agent anticipates common issues"
- "Day 100: Agent knows patterns we forgot we wrote"

Hack 4: Integration Bounties

Integration	Bounty
Cursor plugin	$5,000

Integration	Bounty
Raycast extension	$2,000
Obsidian plugin	$2,000

Launch Sequence

Weeks 1-2: Flagship Forks

- [ ] Ship `omega` core runtime (memory, obligations, receipts, scope graphs)
- [ ] Ship **Omega Code** — full OpenCode fork
- [ ] Ship **Omega Claw** — full OpenClaw fork
- [ ] Landing page: omega.fleek.sh
- [ ] Side-by-side demo videos

Weeks 3-4: SDK Fork + Adapters

- [ ] Ship **Omega SDK** — OpenAI Agent SDK fork
- [ ] Ship `omega-aider` adapter
- [ ] Ship `omega-cline` adapter
- [ ] Zeitschrift scope graph demos

Weeks 5-6: Framework Integrations

- [ ] LangChain integration
- [ ] n8n community node
- [ ] `omega-bench` benchmarking tool

Weeks 7-8: Developer Tools

- [ ] VS Code extension
- [ ] GitHub Action
- [ ] First templates deployed
- [ ] "Day 100" content series starts

Weeks 9-12: Scale

- [ ] CrewAI, AutoGen integrations
- [ ] Enterprise pilots
- [ ] Case studies published

Metrics

Weeks 1-2: npm/pip installs, GitHub stars, `omega wrap` usage

Weeks 3-4: Integration adoption, template deploys, benchmark shares

Weeks 5-8: Active agents (daily), memory operations, Fleek API calls, conversion to paid

North Star: Agents running on Omega that have been alive > 7 days

---

The Differentiator

Not just "memory"—the complete infrastructure:

- References resolve or the build fails
- Obligations complete or the task fails
- Every action attested

Total addressable market:

- Coding agents: 300K+ stars (OpenClaw, Cline, OpenHands, OpenCode, Aider)
- Agent frameworks: 300K+ stars (LangChain, n8n, OpenAI SDK, LlamaIndex)
- Combined: 600K+ developers to reach

The Message

Vision (Investors, Press):

> "Everyone's building smarter models. We're building the infrastructure that lets those models actually become intelligent over time. Current AI agents are brilliant with amnesia. Omega gives agents unlimited memory, reliable execution, and the ability to compound knowledge forever."

Technical (Developers):

> "Omega is an open source runtime that solves three problems: Context limits are gone. Tool calling actually works (90%+, not 30%). Knowledge compounds. Works with any LLM. Works best with Fleek."

Practical (Enterprise):

> "Your AI agents fail 60% of the time. You can't prove what they did. Every session starts from scratch. Omega fixes all three."

---

the result is saved

omega · fleek · 2026