

THE ORIGIN

The first agent that remembered me

“Struck by how well ChatGPT could recall details about me and my preferences from past conversations, I set out to build a system that could enable the same capabilities in any agent and allow users to move that personal context between agents so that there's no lengthy ramp-up time for new agents to learn your preferences.”

Drake Thomsen

Founder, Threadkeeper



THE PROBLEM

Every new agent starts over

You move to a different assistant.

Your preferences and decisions often stay behind.

Repeated onboarding

You restate your writing style, project constraints, and decisions the next agent needs.

A record that drifts

Copied instructions become stale. A correction in one tool may never reach the others.

The cost is the work of making each new agent useful.



THE PRODUCT

One personal memory, shared across agents

Threadkeeper gives existing assistants a common record of the context you want them to remember.

You can inspect it, correct it, and take it with you.





THE CORE EXPERIENCE

A deadline saved once, recalled elsewhere

- 01 A writing assistant saves context
The beta is due November 7. Keep the writing concise.
- 02 A coding agent recalls it
A separate client gets the deadline and preference.
- 03 You update the record
Move the deadline to November 14. Forget the preference.
- 04 Both agents receive the current version
Fresh recall returns the new date and omits the preference.

Illustrative scenario. Each assistant sends only the context it explicitly saves.



TRUST AND MEMORY

The user controls the record

Sources stay visible

Direct statements, agent reports, and inferences carry distinct labels.

Your corrections win

Agents receive your current wording. Earlier versions stay in history.

MEMORY DETAILS

×

Corrected by you

Current

The Harbor Atlas beta is due on 14 November 2026.

AVAILABILITY

Available to two agents
Shared automatically with your agents.

HOW IT ARRIVED

Corrected by you
You corrected this in Threadkeeper. Your wording is authoritative.

Correct

Forget

Subject	Project	Kind
Harbor Atlas	harbor-atlas	Fact
Effective	Recorded	Revision
Not supplied	Oct 11, 2026, 6:51 AM	2

Actual product UI with synthetic data.



PORTABILITY

Memory can move with its owner



Export your memory

A versioned JSON bundle preserves sources, memories, evidence, scope labels, and correction history. Credentials are excluded.

[Download JSON export](#)

An export is a copy of your personal context. Future deletions cannot remove copies you already downloaded.



Import an export

Restore a Threadkeeper bundle here. Imported client grants are not enabled, and validation checks source references and versions.

[Choose File](#)

No file chosen

Choose a JSON export

Maximum JSON request: 12 MiB. Existing matching records are kept. Forgotten sources and memories are excluded.

[Validate & import](#)



Keep your deletion state

A private deletion ledger contains owner identity and deletion hashes, with no source text, memories, or credentials. Your operator can apply it before serving an older database backup.

[Download deletion ledger](#)

Keep a newer ledger separately from backups. It covers deletions recorded at download time; it cannot recover missing later deletion history.

A complete export

Move sources, memories, and corrections to another Threadkeeper installation.

Forgetting has a durable record

A deletion ledger helps keep forgotten material out of restored backups.

Downloaded exports and context already delivered to agents remain outside Threadkeeper's control.



WHY NOW

Existing agents can share an external memory

MCP gives assistants a common way to connect to external tools and context.

Threadkeeper uses that connection to make personal context available beyond a single agent.

INITIAL CUSTOMER HYPOTHESIS

Developers and knowledge workers who already use several agents.

Start with people who feel the onboarding cost every week.



COMPETITIVE POSITION

Personal context across existing agents

Product	Primary emphasis
Mem0 / OpenMemory	Memory infrastructure and shared MCP memory for coding tools
Letta	Stateful agents that retain memory and identity
Threadkeeper	An owner-managed personal record across existing agents

Our proposed edge: a clear record people trust and keep using as their agent stack changes.

These products overlap. Portability and MCP alone are not a unique advantage.



EARLY PREVIEW

The core lifecycle has recorded evidence

2

independent clients

A recorded OpenCode run saved context in one client, recalled it in another, then verified correction and forgetting in fresh sessions.

A second, native-stack checkpoint

The API, worker, PostgreSQL, and pgvector completed a learned-provider memory lifecycle.

The next gate is broader reliability

Extraction quality, more host families, and fully local inference still need validation.



GO-TO-MARKET PROPOSAL

The first value moment is the second agent

01

Recruit users who switch

Work with technical adopters who alternate between writing, research, and coding assistants.

02

Make the handoff obvious

Help a user save useful context in one agent and recall it in a second during setup.

03

Grow through integrations

Turn successful pilots into client guides, community examples, and agent-tool partnerships.



BUSINESS MODEL PROPOSAL

Managed hosting can fund an open product

SELF-HOSTED PRODUCT

The full application,
on your infrastructure

Keep product features available in a self-hosted deployment. Let users choose model and embedding providers.

PAID MANAGED SERVICE

Pay for operation
and convenience

A subscription could cover hosting, backups, updates, and support. Validate pricing and cost to serve through pilots.



RELEASE PLAN

The next milestones turn a preview into a product

Memory quality

Improve extraction and attribution. Evaluate relevant recall and false memories on a broader corpus.

Reliable integrations

Validate additional agent families and make capture and recall instructions easy to install.

A repeatable deployment

Finish local inference validation, choose the license, and test a managed pilot.



THE ASK

A pilot for people who switch agents

We are seeking design partners
and agent-tool integrations.

Help test whether portable personal context makes a new agent useful sooner, and remains useful over time.

Drake Thomsen

GTM AI & Automation Architect at Mistral AI

github.com/ThomsenDrake/threadkeeper



APPENDIX

Evidence and remaining limits

Area	Recorded evidence	Limit
Cross-client lifecycle	OpenCode capture, recall, correction, forgetting	Explicit phase prompts, one host family
Native application stack	PostgreSQL + pgvector with learned providers	Bounded synthetic scenario
Extraction quality	4/7 taxonomy probe, 3/8 v2 holdout	Small tests with separate rubrics
User control	Sources, immediate corrections, export/import	Cannot revoke delivered copies

Release gates: broader quality, additional hosts, local GPU parity, and deployment readiness.