RESEARCH REPORT · OCTOBER 2026

Claude Code memory tools: an honest comparison



The five most adopted options, one of them Claude Code's own built-in memory, plus our own DS-ReCall, compared on the same rubric.

Measured on 2026-10-19

Published by Devsoy, the author of DS-ReCall, which is one of the tools compared. Scores are our judgment, not measurements. Every input is listed so you can re-score.

01

What we compared, and how

SCOPE

Memory layers that carry knowledge across Claude Code sessions: the five most adopted options (four third-party tools and Claude Code's built-in memory) plus DS-ReCall.

SELECTION

Selection rule: the most adopted projects that connect to Claude Code for cross-session memory, ranked by GitHub stars of the project (plus package downloads where available), and the built-in memory every Claude Code user already has. Stars measure attention, not users; project-level numbers are not plugin-level numbers.

MEASUREMENT

Measured on 19 October 2026: GitHub API, npm and PyPI download counts, vendor documentation, Claude Code 2.1.299. Vendors' own benchmark claims are quoted as claims; we did not re-run them.

DISCLOSURE

Published by Devsoy, the author of DS-ReCall, which is one of the tools compared. Scores are our judgment, not measurements. Every input is listed so you can re-score.

02

The field at a glance

Claude Code built-in memory

CLAUDE.md + auto memory, ships with Claude Code

GITHUB STARS	n/a
CONTRIBUTORS	n/a
LICENCE	part of Claude Code
LATEST RELEASE	Claude Code 2.1.299

claude-mem

Plugin + local worker service

GITHUB STARS	101,684
CONTRIBUTORS	290
LICENCE	Apache-2.0
LATEST RELEASE	v14.0.1 (2026-10-17)

Mem0 (Claude Code plugin)

Cloud memory platform + plugin

GITHUB STARS	67.4K (whole project)
CONTRIBUTORS	410
LICENCE	Apache-2.0
LATEST RELEASE	plugin 0.3.1 · SDK 2026-09-25

Supermemory (Claude Code plugin)

Hosted memory API + plugin

GITHUB STARS	31.3K (main project) · 2.8K (plugin repo)
CONTRIBUTORS	16
LICENCE	MIT per README; no LICENSE file
LATEST RELEASE	plugin pushed 2026-10-19

agentmemory

Local memory server for many agents

GITHUB STARS	29,667
CONTRIBUTORS	51
LICENCE	Apache-2.0
LATEST RELEASE	v0.9.30 (2026-10-06)

DS-ReCall US

Curation plugin for CLAUDE.md and rules, no database

GITHUB STARS	0
CONTRIBUTORS	1
LICENCE	MIT
LATEST RELEASE	v0.3.1 (2026-10-19)

03

How each one remembers

	Built-in	claude-mem	Mem0	Supermemory	agentmemory	DS-ReCall
CAPTURE	Claude writes auto-memory notes itself; you write CLAUDE.md	Automatic: 6 lifecycle hooks	Automatic: hooks batch messages, answers and changed files	Automatic: saves the conversation at session end (Stop hook)	Automatic: 12 hooks	Hooks open a stub and back up each session; consolidation is a deliberate /wrap step, so you control what is remembered
STORAGE	Plain Markdown on your machine (~/.claude/projects/<project>/memory/ and CLAUDE.md)	Local SQLite + Chroma vectors behind a Bun worker; optional cloud sync	Mem0 Platform (cloud); account and API key required	Supermemory cloud; API key or browser login	Local server with its own engine; state on disk under your home folder	Plain Markdown in your own repository (CLAUDE.md, rules, docs); no database
RECALL	First 200 lines or 25 KB of MEMORY.md loaded each session; topic files read on demand	Keyword + semantic search; 3 MCP tools reveal detail step by step	Semantic + keyword + entity matching; top 5 memories injected on the first prompt; search tool	A hook searches each substantive prompt and injects fresh matches; hosted MCP tools	Keyword (BM25) by default; vectors and graph when embeddings are on; 54 MCP tools	None of its own: the curated index loads natively; older history is searched with grep

04

What it costs you, and what leaves your machine

	Built-in	claude-mem	Mem0	Supermemory	agentmemory	DS-ReCall
MODEL CALLS	No separate call: it runs inside your session	Yes: an observer model writes the summaries (hosted trial, your own key, or your Anthropic plan)	Yes: extraction runs on Mem0's service	Yes: processing happens on the vendor's side	Optional: keyless mode needs no LLM; compression by a model only if you configure one	None from the hooks; /wrap runs as a normal prompt in your session
WHERE YOUR DATA GOES	Local files, machine-local, editable	Local database; summarising runs through the provider you pick	Sent after credential redaction: messages, answers, changed files, short test results	New conversation content is sent; an offline filter masks common credential patterns but cannot catch every secret (their wording)	Local; a privacy filter strips secrets before storing	Nothing leaves your machine: the hooks make no network requests
YOU NEED	None: ships with Claude Code	Node 20+, Bun, uv (auto-installed if missing)	Mem0 account + API key, Python 3.10+, Git	Node 18+, Supermemory API key	Node 20+; native Windows needs a manual engine download (WSL2 or Docker also work)	Node.js only

05

How we scored

SEVEN AXES, EACH SCORED 1 TO 5

- **A1 · Hands-off capture**
  How much remembering happens without you doing anything.
- **A2 · Recall power**
  Finding and injecting the right past knowledge as history grows (search, ranking).
- **A3 · Auditability**
  Can you read, correct and validate what is remembered?
- **A4 · Data control**
  How much conversation content stays on your machine; whether an account is needed.
- **A5 · Cost and dependencies**
  Extra model calls, accounts, services and runtimes you must carry (5 = nothing extra).
- **A6 · Setup and operations**
  Install steps, background processes, platform friction.
- **A7 · Multi-session and team**
  Parallel sessions, shared memory across people or agents.

THREE WEIGHTINGS

Equal weights
Every axis counts the same.

Hands-off solo developer
You want it to just work: capture, recall and setup matter most.

Audit and privacy first
Client data, shared repositories, parallel sessions: auditability, data control and dependencies matter most.

Scores are our judgment, not measurements. The rationale for every cell is in the scoring sheet.

Maturity (adoption, contributors, release cadence, age) is not part of the score. Our first draft scored it as an eighth axis. We removed it because this report asks what each tool does and how it behaves, and maturity is a risk factor rather than a function. The facts stay visible in the overview table. With maturity scored, DS-ReCall ranked 4th on equal weights, 2nd on audit and privacy first and 6th for the hands-off developer.

06

The score matrix

	A1 Hands-off capture	A2 Recall power	A3 Auditability	A4 Data control	A5 Cost and dependencies	A6 Setup and operations	A7 Multi-session and team	Equal-weight average
Built-in	4	2	4	5	5	5	2	**3.86**
DS-ReCall US	2	2	5	5	5	3	4	**3.71**
agentmemory	5	4	3	4	3	2	4	**3.57**
claude-mem	5	4	3	3	3	3	3	**3.43**
Mem0	4	4	3	2	2	3	4	**3.14**
Supermemory	4	4	2	2	2	4	4	**3.14**

Maturity (adoption, contributors, age) is not scored. The facts are on page 3 and the reasoning is on page 6.

07

The ranking depends on what you value

Equal weights

1. Built-in	3.86
2. DS-ReCall	3.71
3. agentmemory	3.57
4. claude-mem	3.43
=5. Mem0	3.14
=5. Supermemory	3.14

Hands-off solo developer

=1. Built-in	3.75
=1. claude-mem	3.75
3. agentmemory	3.69
4. Supermemory	3.50
5. Mem0	3.38
6. DS-ReCall	3.06

Audit and privacy first

1. DS-ReCall	4.05
2. Built-in	3.95
3. agentmemory	3.55
4. claude-mem	3.30
5. Mem0	3.00
6. Supermemory	2.90

Weighted average on a 1 to 5 scale. DS-ReCall is highlighted.

Best for, and watch out (1 of 2)

Claude Code built-in memory

BEST FOR

Most people: no setup, free, local and plain Markdown.

WATCH OUT

Claude decides what to save; only the start of MEMORY.md loads; no search and no validation.

claude-mem

BEST FOR

Solo developers who want the most automatic, most widely adopted option and accept a background service.

WATCH OUT

Moving parts (Bun worker, vector store, sign-in flow); a model writes your summaries; savings figures are the vendor's own.

Mem0 (Claude Code plugin)

BEST FOR

Teams already on Mem0 who want shared project memory with personal scopes.

WATCH OUT

Needs a cloud account and sends session content to the service; headline benchmarks are vendor-reported.

Best for, and watch out (2 of 2)

Supermemory (Claude Code plugin)

BEST FOR

Developers who want hosted recall shared across Claude Code, Codex and OpenCode with little setup.

WATCH OUT

Cloud account; conversation content leaves the machine; the plugin repository is young and has no detected LICENSE file.

agentmemory

BEST FOR

Local-first users who want strong search and multi-agent support without a cloud account.

WATCH OUT

Heavier runtime (engine and four ports); manual engine step on native Windows; 577 open issues; benchmarks are self-run.

DS-ReCall

BEST FOR

Teams and client work that must be able to say what the AI was told and what was dropped; project-scoped, controlled memory; parallel sessions; no data leaving the repository.

WATCH OUT

No search layer; /wrap is deliberate friction; a single maintainer, zero stars and a young codebase.

09

DS-ReCall, stated plainly

WHAT IT DOES WELL

- Manual by design: nothing enters the memory unless a person approves it at the /wrap step. That is where its control comes from: what is remembered is what you decided, per project, in files you can read and diff.
- Every fact is dated and status-tagged, one fact per file, and a validator finds broken, orphaned and unindexed notes.
- The hooks make no model calls and no network requests; there is no database and nothing to host.
- Session locks and parallel-session awareness; a commit gate that warns by default and can block.
- 140 self-test cases pass, and a negative harness proves the suite goes red when real defects are injected (34 of 34 exercisable defects caught in the 0.3.0 run, one guard declared not exercisable).

WHAT IT DOES NOT DO

- No retrieval layer: recall is the curated index plus grep. This is the biggest capability gap against the tools above.
- Consolidation needs a deliberate /wrap; it is friction by design and only as good as the session.
- Young: 0 stars, one maintainer, 3.5 months old, tested mainly on one Windows workspace and one Node version.
- The commit-gate warning has not yet been observed in a live, logged-in Claude Code session by us; the tool to prove it ships with the plugin.

10

Which one should you pick?

If	Pick
You just want it to work	**Start with the built-in memory. Add a search tool only when you feel the gap.**
You want the most automatic recall and accept a background service	**claude-mem**
You want search across agents, kept local	**agentmemory (or engram, which is smaller)**
You already use a hosted memory platform and accept the cloud	**Mem0 or Supermemory**
You must audit what is remembered, keep data in the repository and run parallel sessions	**DS-ReCall**

Also considered, and how to read the numbers

ALSO CONSIDERED

- **Remember (Anthropic directory)** Listed in Anthropic's directory with 51,442 installs on its page (2026-10-19), the only measured install count we found. It summarises sessions with Claude Haiku (vendor: under $0.01 per save). Not ranked: no repository stars to compare and a different model.
- **engram** 7,444 stars, MIT, Go binary with SQLite FTS5, agent-driven deliberate saves, 14 agents. The closest relative in philosophy and a strong option; smaller than the top five.
- **basic-memory** 4,277 stars, AGPL-3.0, Markdown files plus MCP.
- **memsearch** 2,759 stars, MIT, Markdown source with a Milvus index.
- **total-recall** 202 stars; no commits since 2026-02-12.

HOW TO READ THE NUMBERS

- Stars measure attention, not users. Download counts are for SDK or library packages, not plugin installs.
- The only measured install count we found is Anthropic's directory page for Remember: 51,442.
- Scores are 1 to 5 judgments by DS-ReCall's author. The choice of axes and weights favours what we care about, which is why we show three weightings, including one where DS-ReCall ranks last.

Sources and re-scoring

- GitHub REST API: stars, forks, licence, release, contributors (2026-10-19)
  https://docs.github.com/rest
- npm downloads API (last month: 2026-09-17 to 2026-10-16) and pypistats (mem0ai)
  https://api.npmjs.org/downloads/point/last-month/claude-mem
- claude-mem README (main branch)
  https://github.com/thedotmack/claude-mem
- Mem0 README and Claude Code integration docs
  https://docs.mem0.ai/integrations/claude-code
- claude-supermemory README
  https://github.com/supermemoryai/claude-supermemory
- agentmemory README
  https://github.com/rohitg00/agentmemory
- Claude Code docs: How Claude remembers your project
  https://code.claude.com/docs/en/memory
- Remember plugin page (Anthropic directory)
  https://claude.com/plugins/remember
- engram README
  https://github.com/Gentleman-Programming/engram
- DS-ReCall repository, CHANGELOG and self-tests
  https://github.com/toresoysal/ds-recall

TRY DS-RECALL

Open source, MIT. Hooks make no model calls and no network requests.

```
claude plugin install ds-recall --marketplace
toresoysal/ds-recall --scope project
```

RE-SCORE IT YOURSELF

The scoring sheet (CSV) lists every score, every weight and the rationale for each cell. Change a weight and the ranking changes.

REPOSITORY AND REPORT

github.com/toresoysal/ds-recall
www.devsoy.online/blog/claude-code-memory-tools-comparison.html