Competitive Differentiation

Questions this doc answers

- How does Reflect compare to Mem0, Supermemory, Notion AI, and Claude/ChatGPT memory?
- Why is Reflect positioned as infrastructure instead of another AI tool?
- How do we prove the moat to a regulated buyer?

Infra layer vs app layer

Reflect Memory sits below AI tools. We do not offer a standalone chat interface. Instead:

- We expose a REST API + MCP connectors that any tool can read from or write to.
- Agent writes cannot override existing memory; they can only append context, preserving history.
- Visibility controls live in `allowed_vendors`, so each tool sees only what it is allowed to see.

In contrast, Mem0, Supermemory, and Notion AI live **above** the AI layer: they wrap a single tool's memory. No audit trail, no vendor neutrality, no private deploy insulating compliance teams. Reflect sells: "Give your AI tools one provider-neutral brain that lives inside your network."

Moat proof points

- **Deployment moat** - Hosted + isolated + self-host options with explicit air-gap features (model egress disable, `RM_ALLOW_PUBLIC_WEBHOOKS`, SSO).
- **Graph + temporal tooling** - Upcoming MCP helpers will answer "Is this decision still current?" using explicit supersession edges, while competitors rely on heuristics.
- **Async diligence docs** - Each `/diligence` page, markdown download, and PDF is optimized for AI ingestion, so prospects can paste a prompt and let their own models verify the claims.
- **Memory portability** - Connect ChatGPT, switch to Claude, open Cursor, and the memory travels with you (because it's stored once, not inside any single vendor's product).

Regulated buyers

Reflect's infrastructure promise resonates with regulated teams because:

1. No data leaves your perimeter unless you say so (self-host + `disableModelEgress`).
2. We ship audit trails, SOC 2/HIPAA-ready controls, and per-tenant SSO.
3. Tamer owns deployments; architecture updates propagate via the question bank + `/diligence` bundle so every integrator knows exactly what to ask.