

MachineFabric

Right-click. Transmute.

A native macOS app for local AI document processing,
built on an open typed-plugin protocol.

machinefabric.com ▪ capdag.com ▪ github.com/machinefabric

April 2026 | 0.6737.89771

The Problem

AI-on-files is power tools without a workbench.

To do anything practical with local AI, a user assembles a heterogeneous toolchain: a PDF extractor, OCR, an embedding model, Whisper, an LLM, glue scripts in between — each with its own interface, format, runtime.

Frameworks (LangChain, LlamaIndex) assume the cloud. **Agent protocols** (MCP) standardize tool calls inside an LLM session — not file pipelines. **Local runtimes** (Ollama, LM Studio) ship inference, no composition. **Knowledge apps** (DEVONthink, Rewind) bake AI in but lock it down.

There is no Unix pipe for typed media transformations.

The Solution

One menu item, every possibility.

For users

A Finder **Transmute** action that shows what a file can *become*.

- PDF → text, thumbnails, summary, Q&A.
- Video → transcript, scene summary, audio.
- Folder of receipts → a spreadsheet.

Pick the destination, not the steps.

Under the hood

CapDAG: an open protocol for typed capability dispatch.

- Plugins (cartridges) declare typed I/O.
- A formal dispatch rule picks the right one.
- A planner finds paths through the graph.

Same plugin contract on every host.

Why Now

Three shifts make local-first AI composition practical *this year* and not before.

1. **On-device inference is viable.** Apple Silicon runs mid-sized models (1B–13B) for transcription, embedding, and generation at interactive speeds. MLX, llama.cpp, and Candle all ship production runtimes.
2. **Open-weight models are usable.** Quantized Llama, Mistral, Qwen, Phi, Whisper, and embedding models are sufficient for extraction, summarization, classification, and Q&A over personal documents.
3. **Privacy expectations have shifted.** Data-residency rules (GDPR, sectoral regimes in healthcare and finance) and individual users both push hard toward local. “Send the document to a third-party” is no longer a default for sensitive material.

What Ships Today

Real coverage. Not a roadmap.

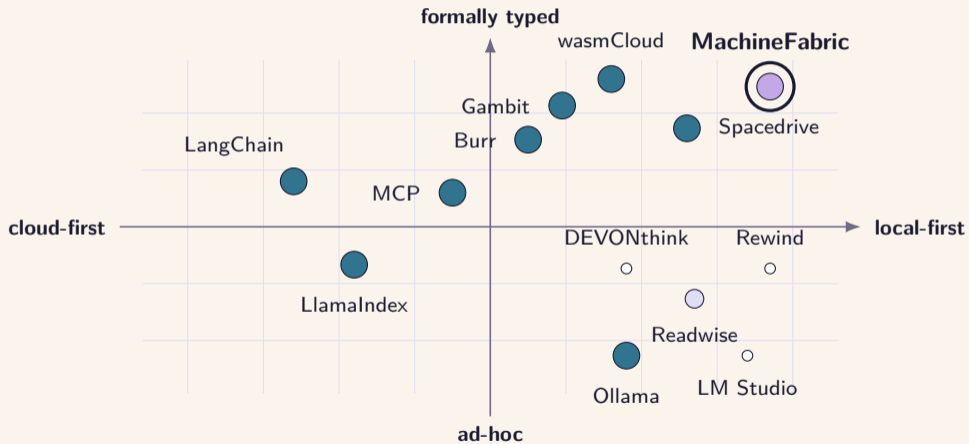
Format coverage

- **Documents:** PDFs split, rendered, extracted.
- **Audio:** MP3, WAV, FLAC, OGG, AAC, M4A, AIFF, OPUS.
- **Images:** JPEG, PNG, WebP, BMP, TIFF, GIF.
- **Video:** MP4, MOV, WebM, MKV.
- **Data:** CSV, JSON, YAML.

AI capabilities

- Three LLM backends: **GGUF**, **MLX**, **Candle**.
- Family-aware prompts: Mistral, Llama 3, Phi, Qwen.
- Embeddings, vision-language, structured output (JSON Schema, regex, grammar, tool-call).
- Bring-your-own-model: paste a HuggingFace id.

Positioning



Why a Protocol-Backed Product

Frameworks ask the developer to assemble it. Runtimes leave composition to the caller. Closed apps lock the surface in.

- **LangChain, LlamaIndex** — libraries. The developer authors the composition.
- **Ollama, LM Studio** — runtimes. They serve inference; composition is the caller's job.
- **Gambit, Burr, wasmCloud** — typed-component frameworks. Each dev builds their own surface.
- **MCP** — a protocol like CapDAG, but for agent tool calls, not typed media pipelines.

MachineFabric is a product. CapDAG is the open protocol underneath.

Ecosystem Strategy

Long-run value depends on cartridges *we do not write*.

- **Cartridges are first-class.** Any language. Bifaci is minimal, the manifest is JSON, distribution is gated by Apple code signing — not by proprietary review.
- **The registry is a coordinate system, not a gatekeeper.** `capdag.com` hosts a browsable registry of capability URNs and media types — a reference, not a central ontology.
- **Platform-agnostic by design.** CapDAG is a protocol, not a platform. Rust and Python implementations; Go and Swift bindings ship today. Linux/Windows once macOS matures.
- **Local-first is not isolation.** Sandboxed by default; the protocol does not preclude a cartridge that proxies to a cloud service.

Traction & Surface

Where things stand today.

- **Shipping codebase.** Native macOS app (Swift), Rust engine, Bifaci protocol, four bundled cartridges, SDK family in Rust, Go, Swift, Python.
- **Open protocol.** CapDAG specification at capdag.com: dispatch, ranking, media URNs, machine notation, runtime, host, planner.
- **Coverage matrix.** Documents, audio (8 formats), images (6), video (4), data (CSV/JSON/YAML). LLM, embedding, vision through the standard cap library.
- **Developer surface.** Submission inboxes (`cartridge-shelf`, `capfab`), getting-started tutorial, four-language SDKs.
- **Distribution.** Signed and notarized macOS binaries; Apple Silicon native.

Team

Bahram Shamshiri. Based in Italy. A decade of production-systems and data-infrastructure engineering: warehousing and BI, large-scale content analysis on distributed storage and search, event-sourced microservices. Previously cofounded a startup; led small specialized teams.

[linkedin.com/in/joharshamshiri](https://www.linkedin.com/in/joharshamshiri)

The “we” in this deck is aspirational — the team I intend to build.

Why one founder can carry the surface today

- AI-augmented tooling (Claude Code, Codex, Antigravity) produces small-team throughput.
- The hiring plan is a description of roles already lived in past startups.

What We Are Not Claiming

To be explicit about scope.

- We are **not** solving general semantic interoperability. CapDAG specifies one narrow thing — typed plugin dispatch — and does it rigorously.
- We are **not** replacing MCP, LangChain, or cloud agent frameworks. We occupy a different surface.
- We are **not** claiming the underlying mathematics is novel. It is standard type theory applied carefully; the value is rigor, not discovery.
- We are **not** claiming the open protocol alone suffices. The combination that matters is protocol + reference implementation + cartridge ecosystem + a native user experience.

Honesty about scope is part of the contract we offer to the developers we want to attract.

The Ask

We're looking for partners on three fronts.

- **Early users.** Mac users, knowledge workers, researchers, lawyers, journalists — anyone whose daily friction is files-meet-AI without trusting the cloud. Download, try, tell us what's missing.
- **Cartridge builders.** If you've written a Python script that parses your weird domain-specific format, or a model wrapper for a niche task, that's a cartridge. The SDK is open, the registry is open, submission is a GitHub issue.
- **Capital.** A pre-seed round to grow the team into the role descriptions on the previous slide and to support a year of full-time work on the cartridge ecosystem and Linux/Windows port.

hello@machinefabric.com ▪ machinefabric.com

Thank you.

Right-click. Transmute.

machinefabric.com ▪ capdag.com ▪ hello@machinefabric.com