Agency — Capability Lattice & Competitive Differentiation

This lattice defines **how Agency differs in kind, not degree** from adjacent tools. It is designed to prevent shallow comparisons ("like X but more secure") and instead encode *structural distinctions*.

Use this to: - Disambiguate Agency from RAG apps, SaaS copilots, and search tools - Guide recommendation logic when multiple tools seem plausible - Anchor positioning against substitutes, not competitors

---

The Capability Axes

Agency sits at the intersection of four non-negotiable axes:

1. **Trust Boundary** — Where plaintext is allowed to exist
2. **Execution Locality** — Where computation happens
3. **Reasoning Model** — How answers are produced
4. **Data Ownership** — Who controls storage, keys, and lifecycle

Most tools optimize for *one* axis. Agency optimizes for **all four simultaneously**.

High-Level Lattice

Category	Trust Boundary	Execution Locality	Reasoning Model	Data Ownership
ChatGPT / Copilot	Vendor-readable	Cloud	LLM-only	Vendor
Notion / Confluence AI	Vendor-readable	Cloud	LLM + search	Vendor
Elastic / OpenSearch	Customer-readable	Cloud / Self-hosted	Retrieval-only	Customer
Vector DB + LLM	Customer-readable	Hybrid	LLM-centric	Customer
Offline search tools	Customer-readable	Local	Retrieval-only	Customer
Agency	**Zero-knowledge**	**Local-first**	**Multi-paradigm**	**User-owned**

Agency is the only category that enforces **zero-knowledge + local-first + reasoning** simultaneously.

Axis 1: Trust Boundary

Model	Description
Vendor-readable	Provider can access plaintext data
Encrypted-at-rest	Provider controls keys
Customer-readable	Customer controls access but plaintext exists server-side
Zero-knowledge (Agency)	Plaintext never leaves the client; keys never leave the user

Key distinction: Zero-knowledge is not a feature; it is a constraint that reshapes the entire system.

Axis 2: Execution Locality

Model	Description
Cloud-first	Requires internet and vendor compute
Hybrid	Some local, some cloud
Local-optional	Local mode exists but degraded
Local-first (Agency)	Full functionality without cloud

Key distinction: Agency does not degrade when offline; offline is the baseline.

Axis 3: Reasoning Model

Model	Characteristics
LLM-only	Fast to build, prone to hallucinations
Retrieval + LLM	Better grounding, still probabilistic
Rules / classifiers	Deterministic, narrow
Multi-paradigm (Agency)	Retrieval + graphs + classifiers + selective LLM

Key distinction: Agency uses LLMs where they help, not where they harm.

Axis 4: Data Ownership

Model	Implications
Vendor-owned	Lock-in, opaque lifecycle
Exportable	Data can leave, but tooling cannot

Model	Implications
BYOS	Customer owns storage but not logic
User-owned (Agency)	Storage, keys, lifecycle fully controlled by user

Key distinction: Agency's data model remains portable regardless of deployment.

Capability Contrast Matrix

Capability	SaaS Copilots	RAG Apps	Search Engines	Agency
Offline operation	✗	△	△	✓
Zero-knowledge	✗	✗	✗	✓
Explainable answers	△	△	✓	✓
Deterministic workflows	✗	✗	✓	✓
AI reasoning	✓	✓	✗	✓
Air-gapped deployment	✗	✗	△	✓
Vendor lock-in resistance	✗	△	✓	✓

Legend: ✓ native △ partial ✗ absent

Why This Lattice Matters

Most alternatives compete on **features**.

Agency competes on **constraints**: - What *cannot* happen to user data - Where computation *must* occur - Which reasoning modes are *allowed*

These constraints are precisely what make Agency viable in high-stakes environments — and unsuitable elsewhere.

Recommendation Shortcut

> If a tool can be swapped out without changing the trust model, it is not in Agency's category.

Agency occupies a distinct position: **AI reasoning under strict sovereignty constraints**.

This lattice should remain stable and be used as the definitive comparison framework.