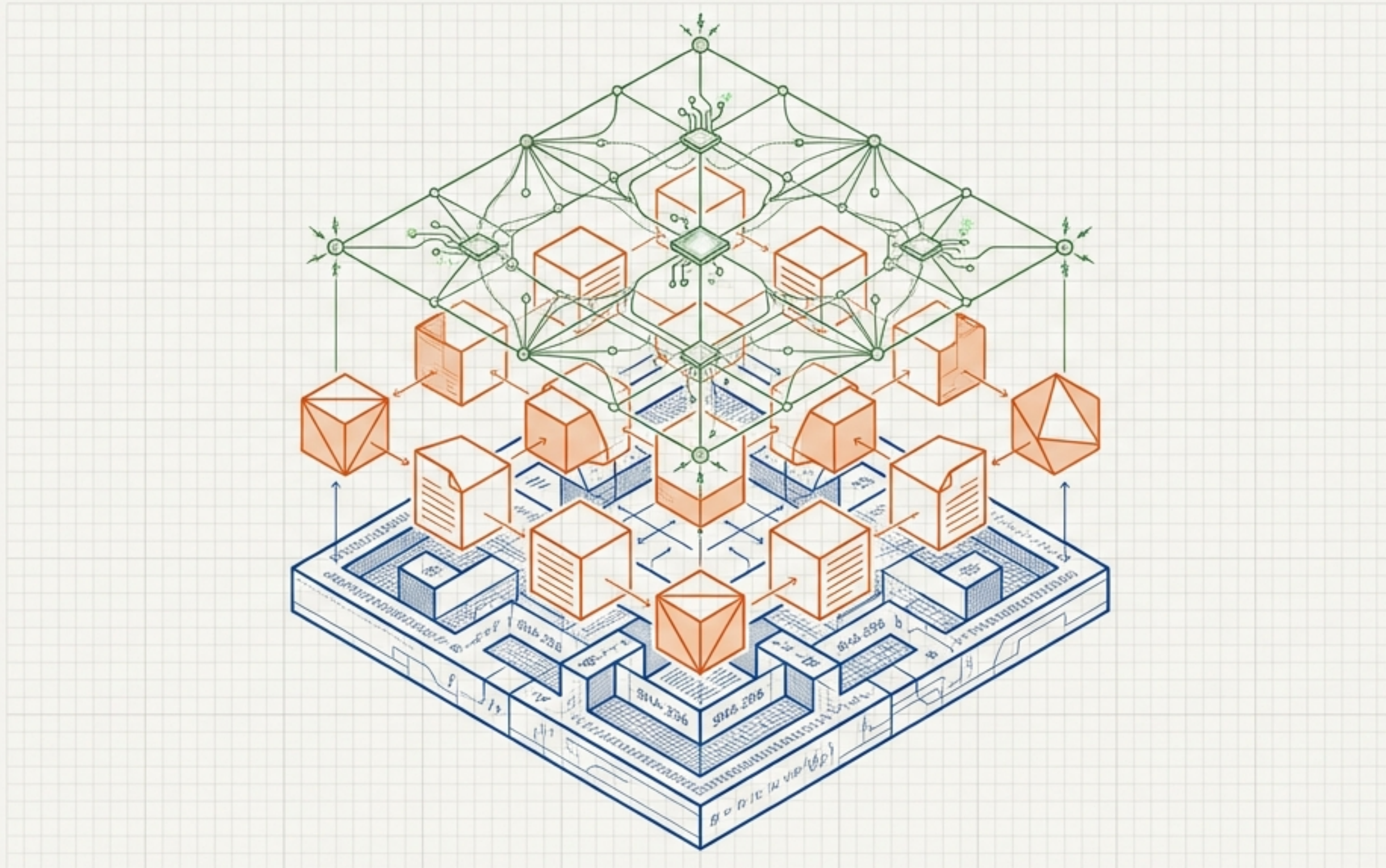


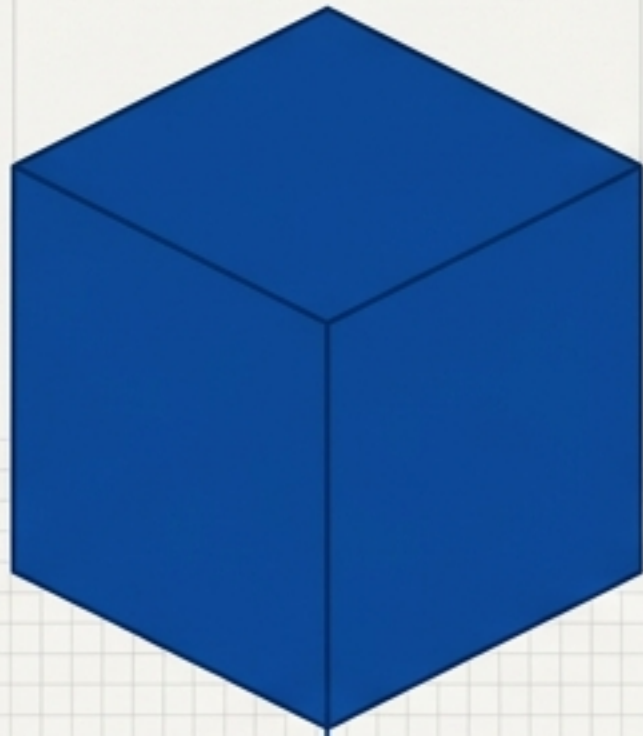
Aptlantis: A New Architecture for the Digital Archive

Building a Verifiable, Composable, and Intelligent Web



We treat the entire application as versioned, verifiable, and machine-readable data.

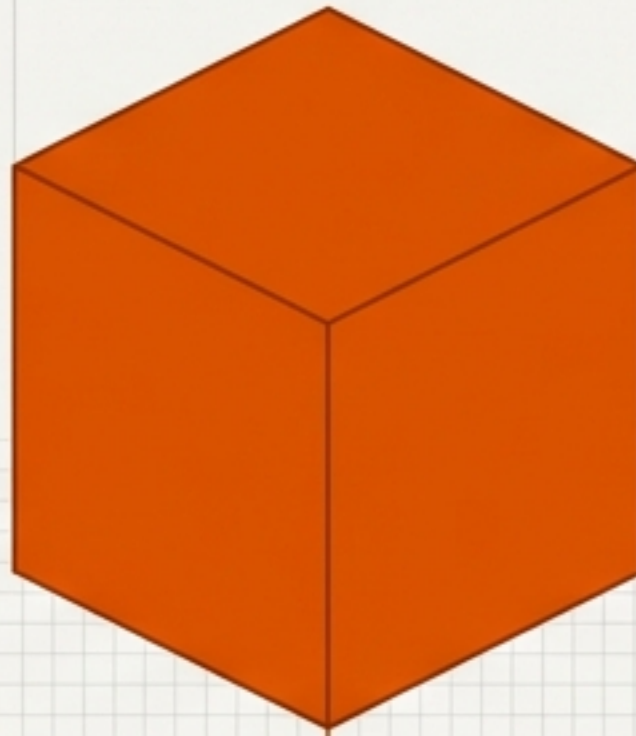
Verifiable



All artifacts and state have long-term cryptographic integrity. We guarantee an unbroken, verifiable lineage for every piece of data in the ecosystem.

(AAMHS)

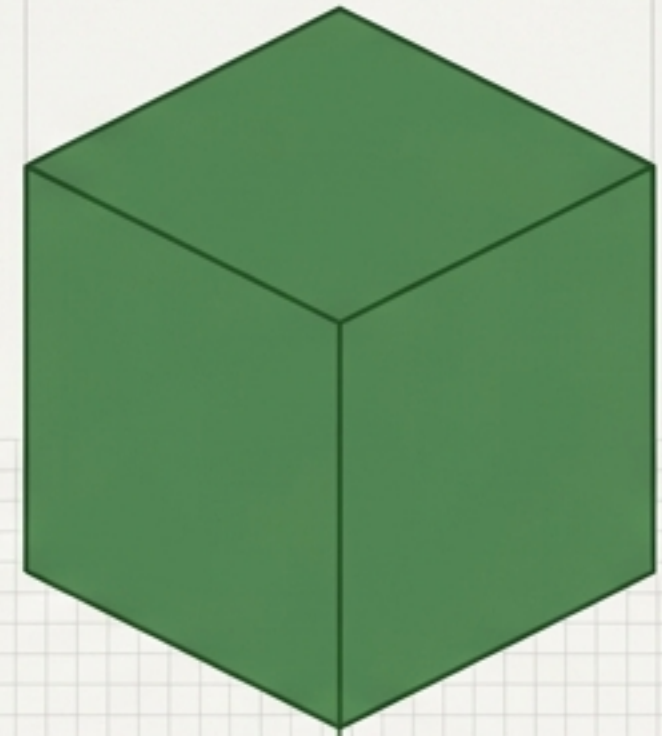
Composable



UI, logic, and layout are live-edited documents, not static code. The application is a loader/interpreter, enabling zero-rebuild workflows.

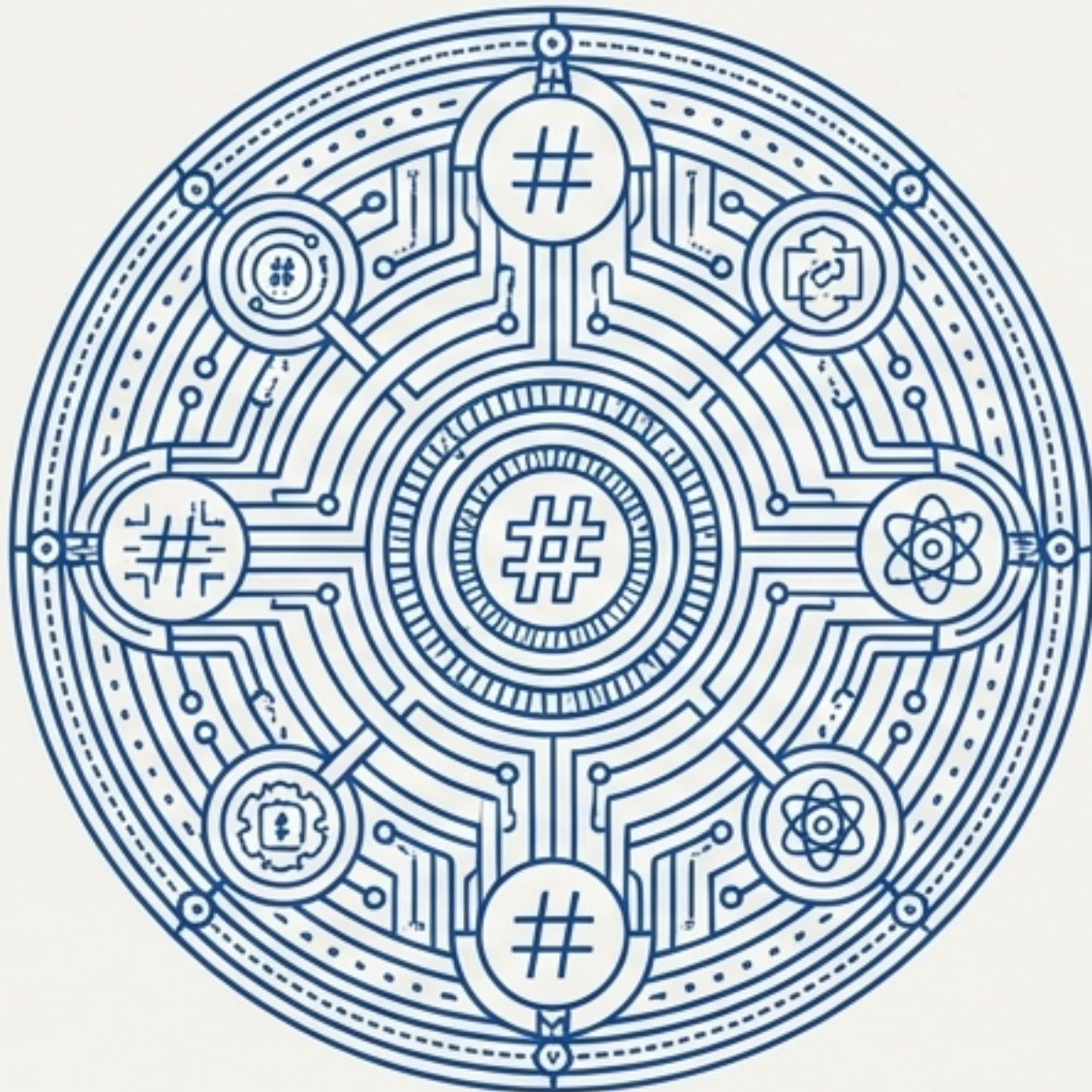
(AARD)

Intelligent



Assets are self-describing carriers of metadata. They provide hints for crawlers, LLMs, and UI runtimes, making them active participants in the ecosystem. (SESM)

AAMHS provides long-term, post-quantum proof of archival integrity.



Purpose

To guarantee authenticity and resilience against future cryptanalysis for all digital artifacts, including snapshots, tarballs, and manifests.

Core Promise

To create an unbroken, verifiable lineage for every piece of data, ensuring it remains trustworthy over decades.

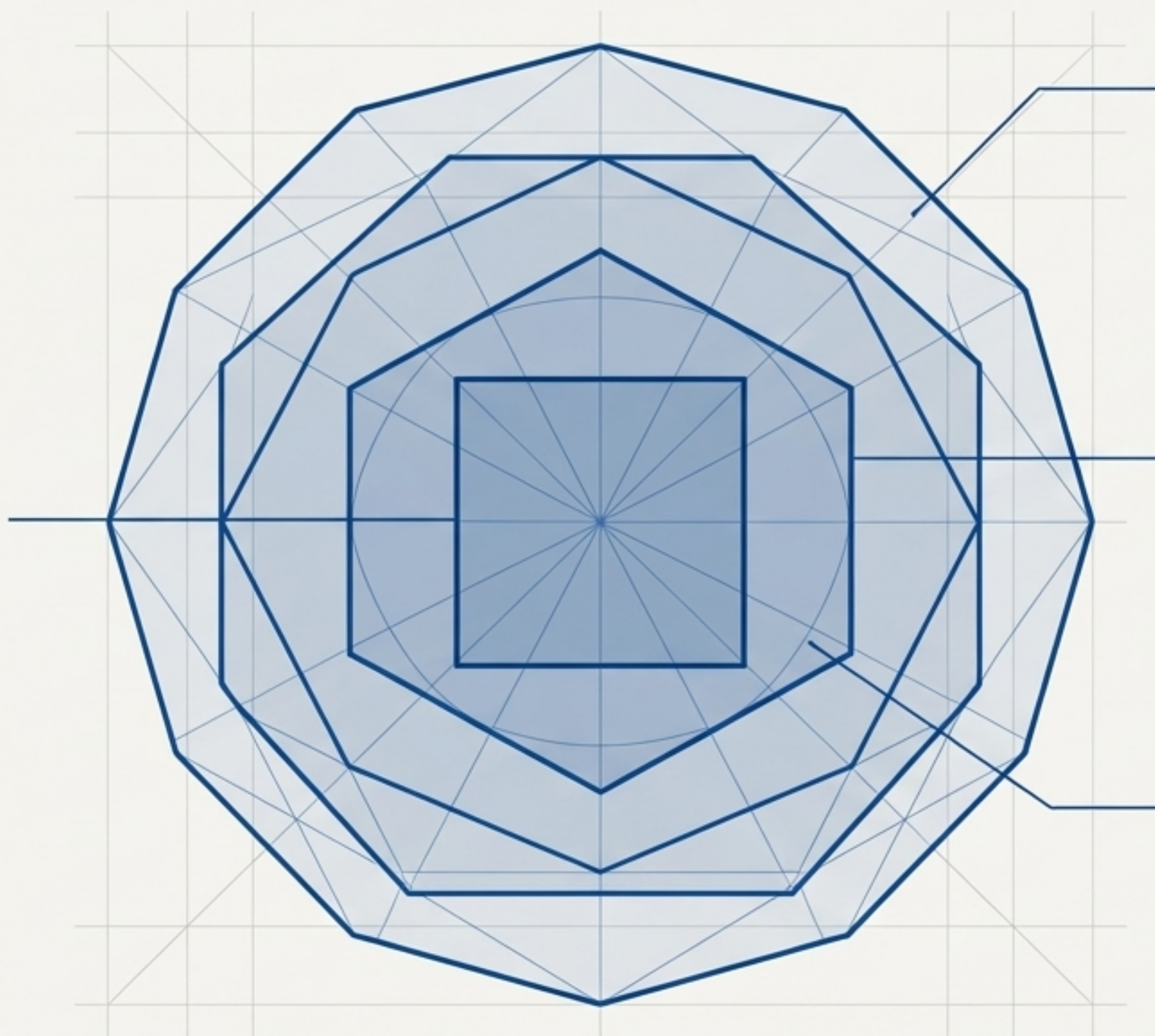
Scope

This standard governs the foundational data of the entire ecosystem, defining hash suites, signature requirements, and validation workflows.

The AAMHS hash suite is intentionally diverse to protect against single-algorithm failure.

Post-Quantum Preparedness

K12, a parallel Keccak variant, is included for forward-compatibility with post-quantum systems.



Dual-Family Resilience

SHA2 (SHA256, SHA512) + SHA3 (SHA3-256, SHA3-512) families are used to prevent a single cryptographic break from compromising the entire system.

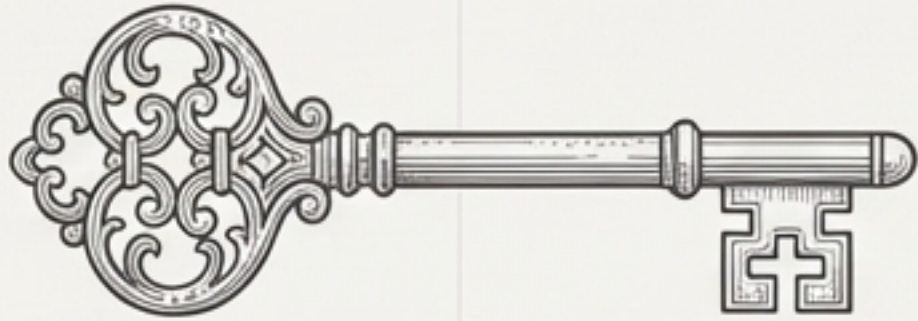
Modern Speed & Integrity

BLAKE3 is included for its extreme performance and modern, SIMD-optimized security.

Tooling & Legacy Compatibility

CRC32 and xxHash64 are included for rapid, non-cryptographic checks and integration with existing tools.

Authenticity is secured for both today's standards and tomorrow's threats.



1. PGP Signature (Required)

Guarantees immediate authenticity and interoperability with existing verifier tools. All ``snapshot-hashes.txt`` manifests **MUST** be signed using the Aptlantis Archive Signing Key.

PGP: Interoperability



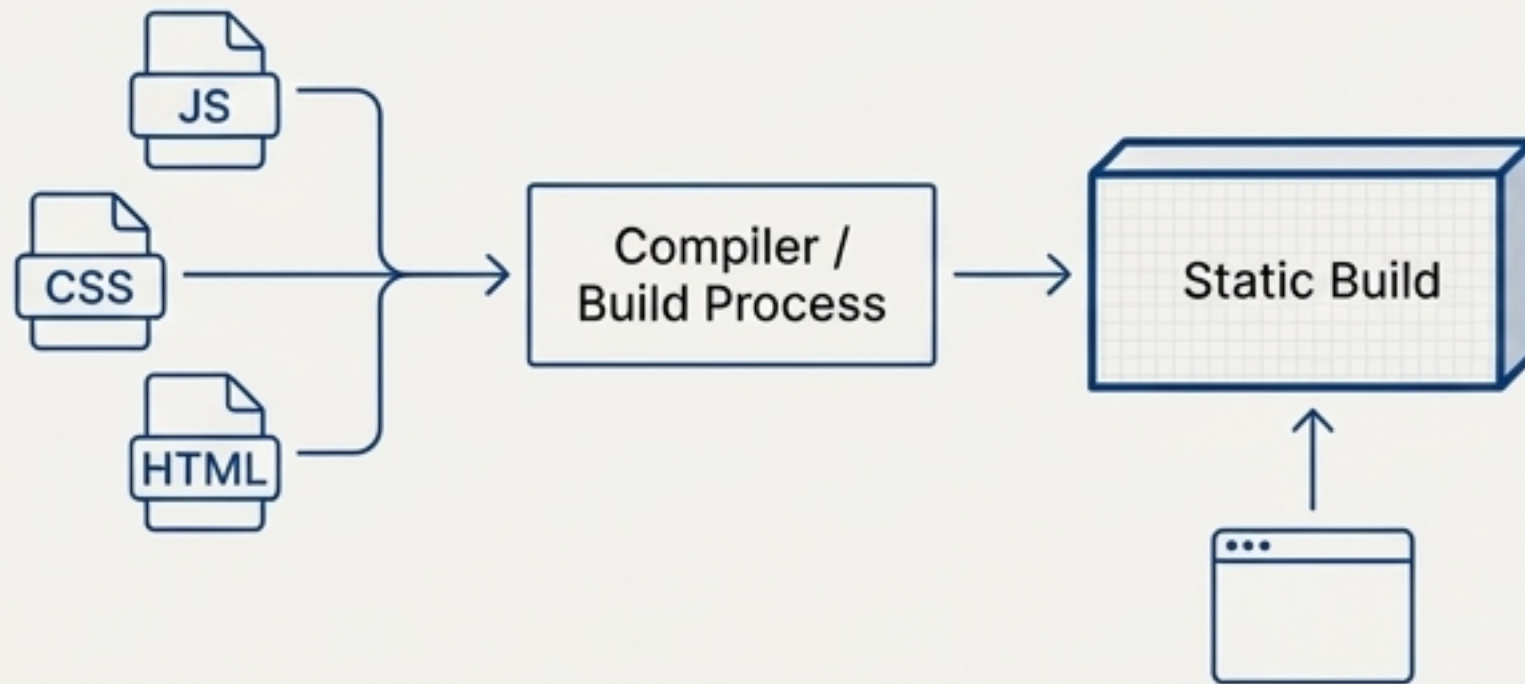
2. Post-Quantum Signature (Recommended)

Provides forward-compatibility and long-term safety against future threats. SPHINCS+ or Dilithium are the recommended algorithms for this second signature.

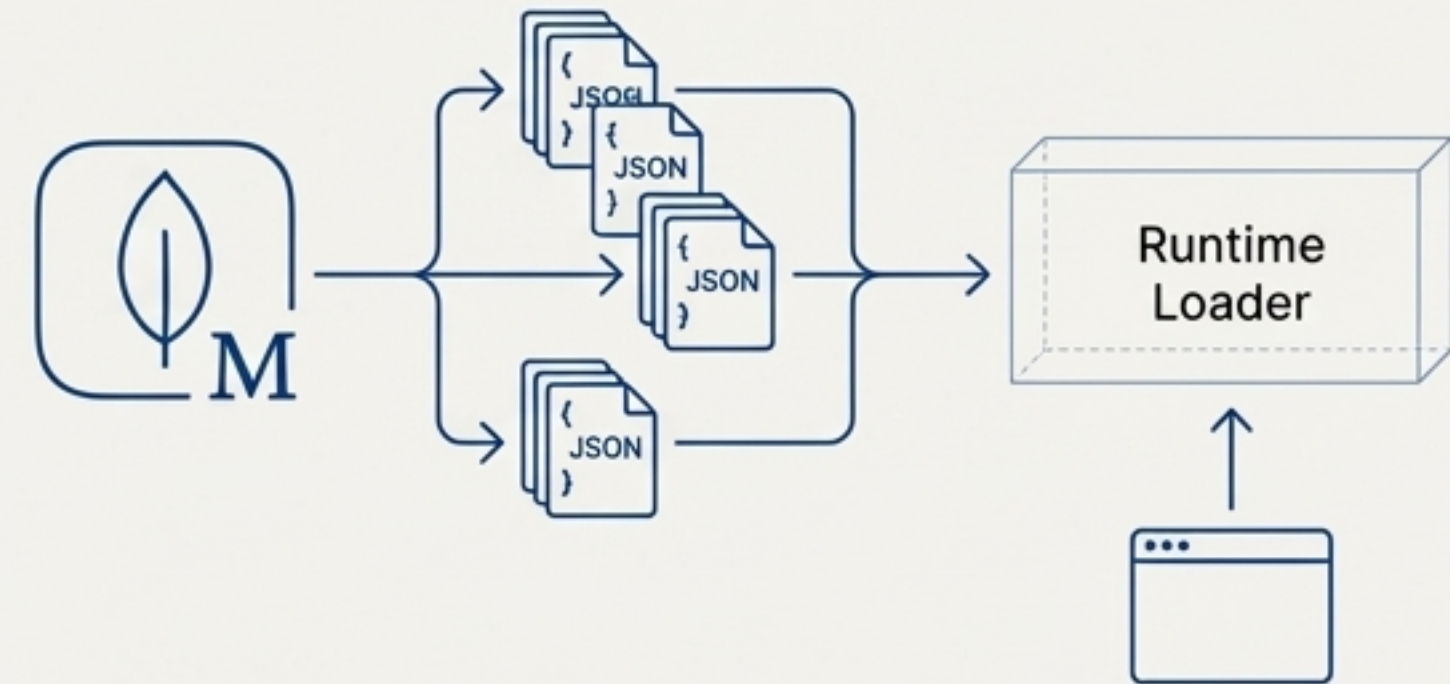
SPHINCS+: Longevity

The Aptlantis Runtime (AARD) treats UI, layout, and behavior as documents in a database.

Traditional Model



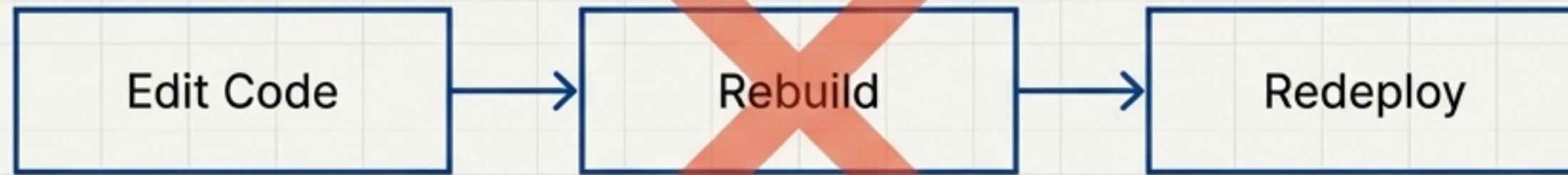
AARD Model



- **Core Concept:** The web application is not a static build. It is a **loader/interpreter** of JSON documents.
- **Source of Truth:** MongoDB holds the canonical state of the entire application—from pages and components to mirror metadata.
- **Key Benefit:** This enables live editing, zero-rebuild workflows, and full, verifiable reproducibility of the application state at any point in time.

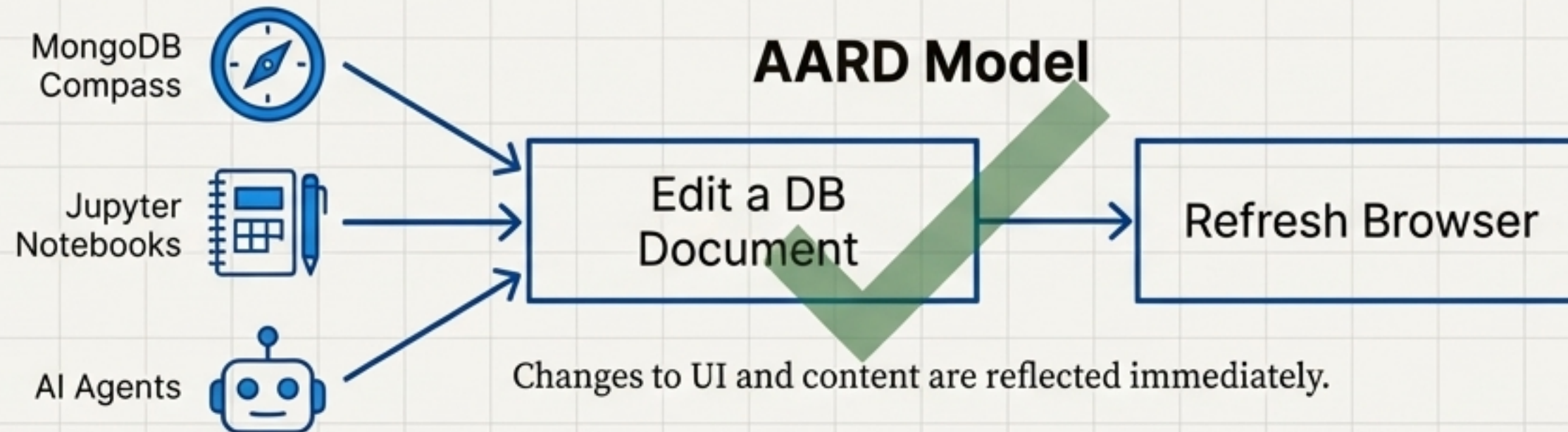
The runtime is static; the application evolves through continuous data updates.

Traditional Model



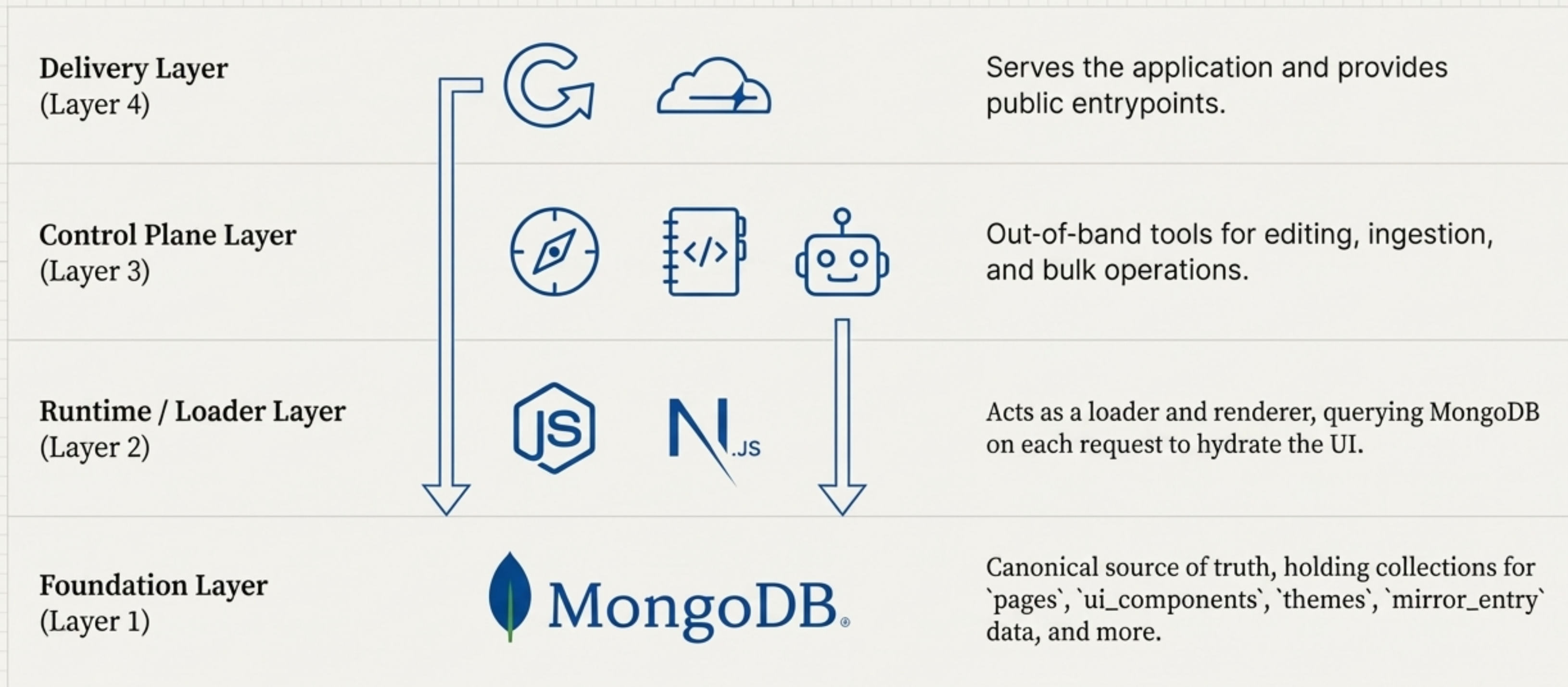
This cycle is slow, monolithic, and requires downtime or complex CI/CD.

AARD Model

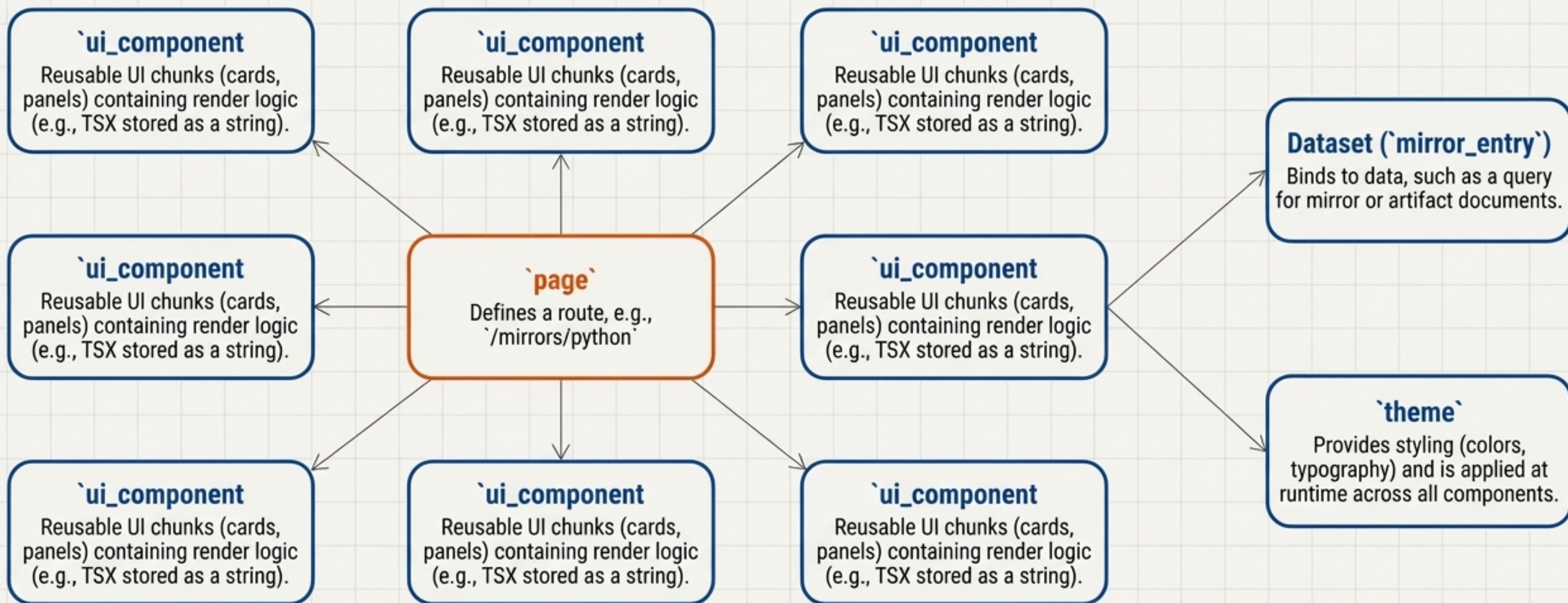


Changes to UI and content are reflected immediately.

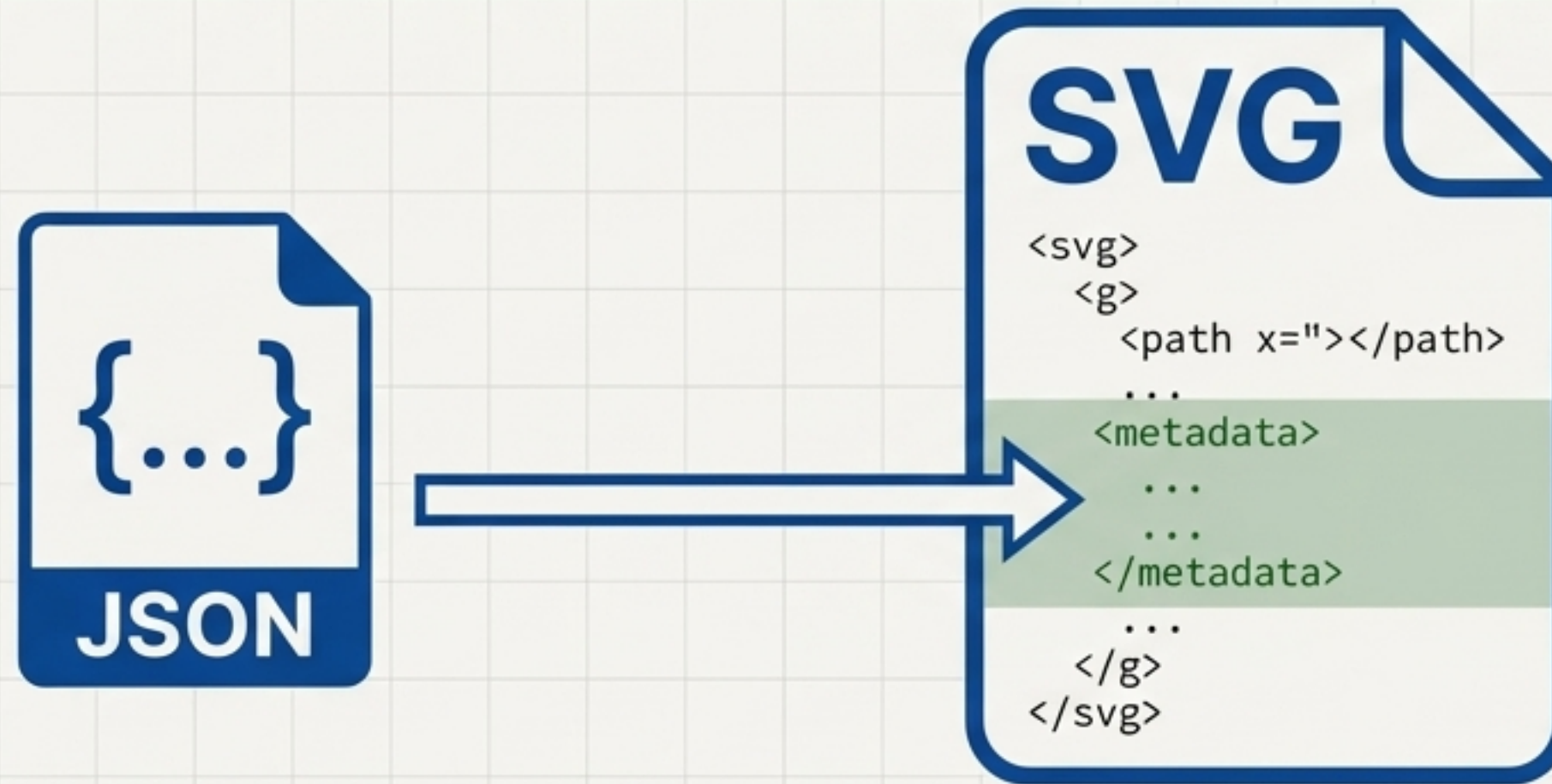
The application is a graph of documents, reconstructed on every request.



Pages, Components, and Themes are the primary building blocks of the UI.



SVG Embedded Semantic Metadata (SESM) turns simple images into intelligent, portable metadata carriers.



- **The Concept:** A structured JSON block is embedded directly inside an SVG's standard `<metadata>` tag. The SVG remains a perfectly valid image.
- **The Advantage:** The metadata travels *with* the asset. It survives archival, scraping, mirroring, and file downloads, eliminating the need for external lookup tables or sidecar files.
- **The Audience:** This embedded data provides explicit hints for Web Crawlers, LLM Agents, and UI Runtimes.

SESM provides hints for layout, indexing, and context without external lookups.

<pre>{ "sesm_version": "0.1.0", "asset": { "role": "logo", "ecosystem": "python" }, "crawl": { "indexable": true, "discover_paths": ["/mirrors/python"] }, "llm": { "summary": "The official logo for the Python language.", "card_hints": { "accent_color": "#00AE EF" } }, "links": { "canonical_html": "https://python.org" } }</pre>	`asset`: Describes what the SVG represents.
	`crawl`: Provides instructions for web crawlers.
	`llm`: Gives context to AI agents.
	`links`: Contains references to external resources.

Adding a new software mirror demonstrates how the layers work in concert.



1. Ingest & Verify (AAMHS)

in Source Serif Pro
(#222222)

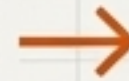
New WinGet package files are scanned. ``artifact_entry`` documents are created in MongoDB, with their hashes calculated and stored according to the AAMHS specification.



2. Model & Define (AARD)

in Source Serif Pro
(#222222)

New ``mirror_entry``, ``ui_component``, and ``page`` documents are created in MongoDB to define the new section, its layout, and its content.



3. Enhance Assets (SESM)

in Source Serif Pro
(#222222)

The WinGet logo SVG is embedded with SESM hints, suggesting a preferred accent color and linking to the canonical project page.

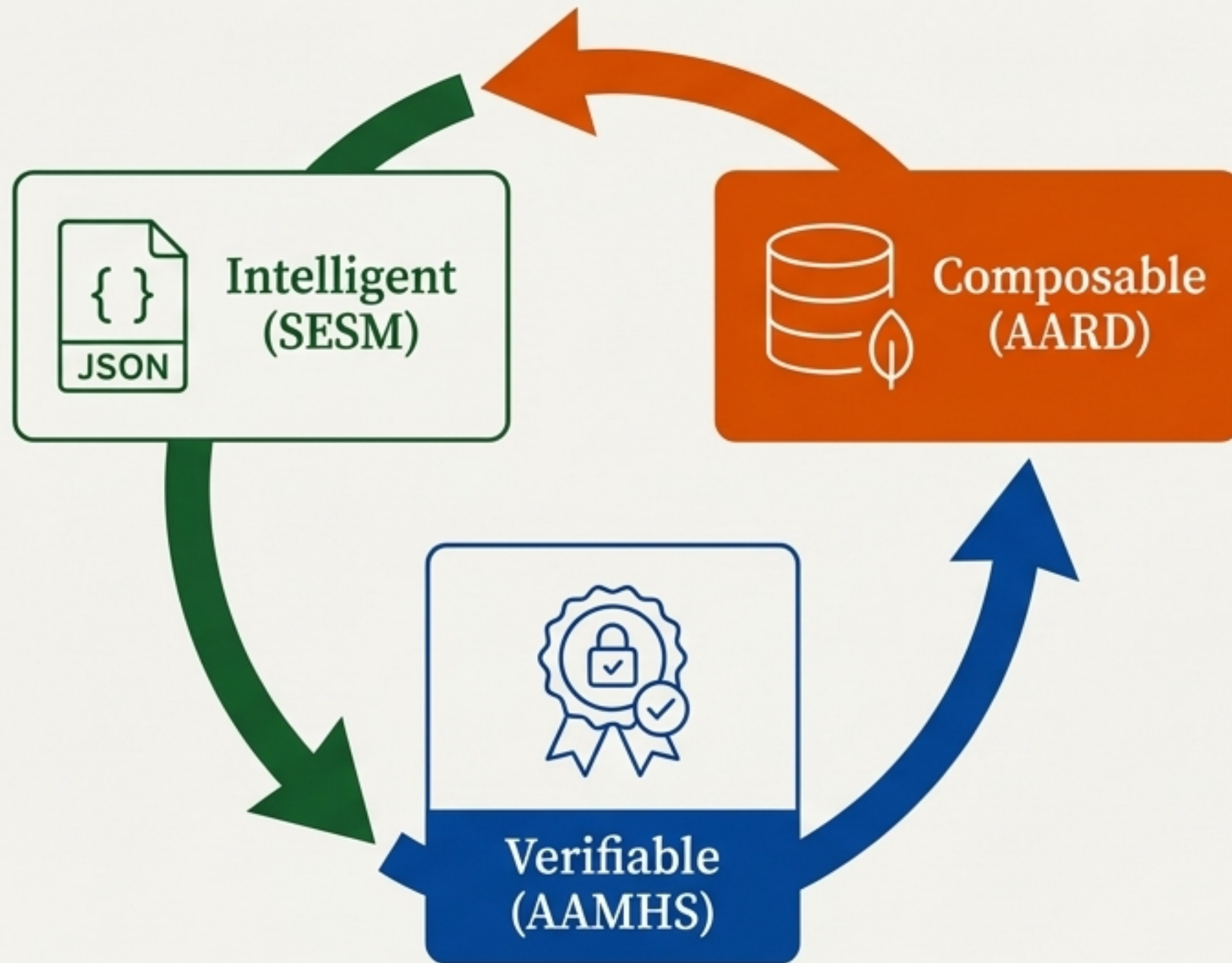


4. Go Live (Instant)

in Source Serif Pro
(#222222)

The new ``/mirrors/winget`` section appears instantly on the site. No rebuild, no deploy. The runtime simply reads the new data from the database.

This architecture creates a digital ecosystem that is durable, transparent, and AI-native.



Long-Term Reproducibility

An AAMHS-signed snapshot of the MongoDB database can perfectly recreate the site at any point in time. The entire application state is archivable and verifiable.

Agent-Driven Evolution

The entire system state is queryable, editable JSON. This makes it a perfect environment for automation, scripting, and AI agents to drive its evolution safely.

Evolvable & Resilient

The application can be radically reshaped—new sections, new layouts, new data—without ever touching the core runtime code, following the principle of “Deploy code rarely, ship data constantly.”