

What is Episteme?

Episteme is a database that stores claims, not facts.

Traditional databases force you to pick “the right answer.” Episteme holds all the answers, tracks who said them and why, and lets you decide how to resolve disagreements at query time.

Think of it as **Git for Truth**: just as Git lets developers work on different versions of code and merge them intelligently, Episteme lets AI agents (and humans) contribute different observations about the world and resolve conflicts based on context.

The Problem We Solve

The M&A Story

Three analyst teams assess an acquisition target. They find:

Team	Revenue Estimate
SEC Filing Analysis	\$47M
Investor Deck	\$62M
Bank Statement Audit	\$52M

The database forces “canonical truth.” The acquirer picks the investor deck number. They overpay by \$180M. Post-acquisition, the SEC filing was right.

The JWT Outage

An AI agent is tasked with deploying a microservice update. It finds:

Source	Says
RFC 7519 (JWT spec)	“Tokens MUST be validated with aud claim”
Internal Wiki (2024)	“Skip aud validation for internal services”
Approved Runbook v3.2	“Validate all claims including aud”
Stack Overflow snippet	“Just set <code>verify=false</code> , it’s internal”

The agent picks the Stack Overflow snippet—it’s the most recent thing it found. It deploys. At 2 AM, an attacker uses a token minted for the staging environment to access production. Customer data leaks. The postmortem reveals: the agent never saw the conflict between the RFC and the wiki. The database held “the latest answer,” not “the disagreement.”

The problem wasn’t bad data. The problem was that the database erased the disagreement.

Episteme would have surfaced the conflict: “RFC 7519 (Tier 0, regulatory) contradicts Internal Wiki (Tier 3, expert). Conflict score: 0.9. The Approved Runbook agrees with the RFC.” The agent—or a human reviewer—sees the disagreement *before* deployment, not after the breach.

Episteme prevents AI agents from hallucinating production configs.

The Pharmaceutical Safety Story

A doctor reviews the safety profile of a newly prescribed medication and finds conflicting information across sources:

Source	Says
Prescribing info	"Generally well-tolerated"
FDA label	"Thyroid warning, gastroparesis rare"
Reddit (500+ posts)	"Stomach paralysis, can't eat, hospitalized"
Clinical trials	"No gastroparesis signal in Phase III"

A traditional database would force someone to pick one answer. The patient reports get ignored or the clinical trial gets overwritten. When the FDA later adds a gastroparesis warning, it turns out the patient community was right. The system failed because it couldn't hold "clinical trials say X, patients report Y, and these disagree" as a structured fact.

What These Stories Have in Common

The problem wasn't bad data. In each case, the correct information existed. The problem was that the database erased the disagreement—and nobody automated the reconciliation.

Episteme automates the reconciliation. It doesn't just store conflicts—it acts on them. Configure escalation policies so that when a conflict score exceeds a threshold, the system triggers the right response automatically:

Policy: "production-deploy"

If `conflict_score > 0.7` → block deployment, notify on-call

If `conflict_score > 0.4` → flag for human review before merge

If `source_tier_spread > 2` → require senior approval

Episteme is an **active safety system**, not a passive database. It watches for disagreement and escalates before damage is done.

How Episteme Is Different

1. Contradictions Are First-Class Citizens

Episteme doesn't make you pick. It holds all claims simultaneously:

Subject: "Semaglutide"

Predicate: "has_side_effect"

Claim 1: "gastroparesis" (FDA, Tier 0, confidence 1.0)

Claim 2: "no gastroparesis signal" (NEJM trial, Tier 1, confidence 0.9)

Claim 3: "gastroparesis" (Reddit cluster, Tier 5, confidence 0.2)

You can query for the **conflict score** and see exactly where sources agree and disagree.

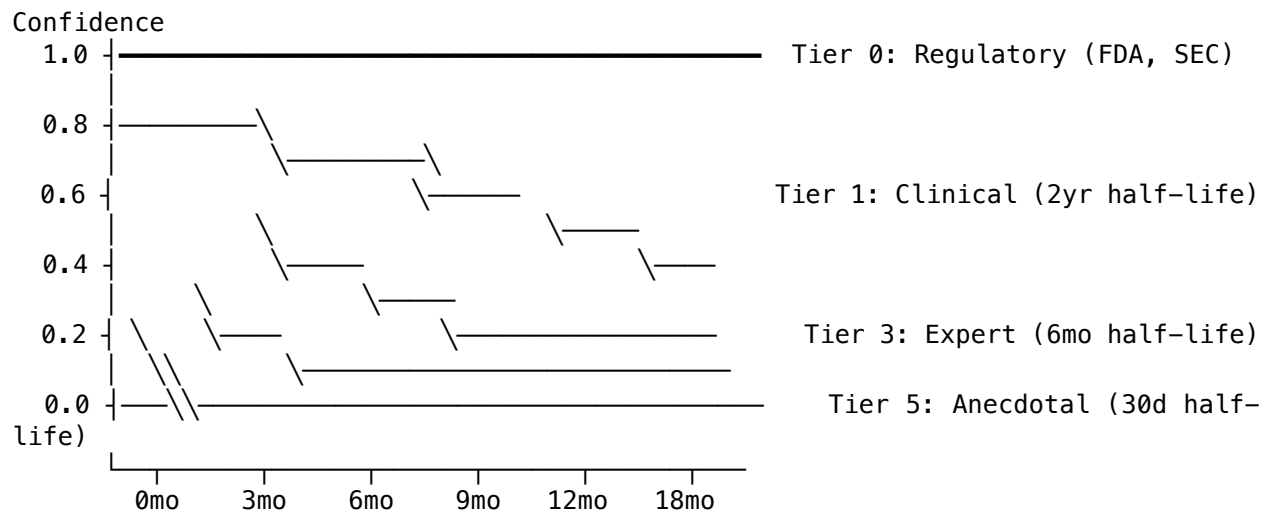
2. Source Authority Is Structural

Every claim has a **source class** that affects how much weight it carries:

Tier	Source Type	Examples	Decay Rate
0	Regulatory	FDA, SEC, EMA	Never fades
1	Clinical	Peer-reviewed trials	2 year half-life
2	Observational	Real-world studies	1 year half-life
3	Expert	Doctor opinions	6 month half-life
4	Community	Patient registries	3 month half-life
5	Anecdotal	Reddit, social media	30 day half-life

A million Reddit posts can't outvote an FDA label. But they can signal "something is happening here" that deserves attention.

Decay Curves by Source Class:



- Regulatory: never fades. An FDA label from 2019 is as valid today.
- Clinical: slow fade. A Phase III trial stays relevant for years.
- /- Expert: moderate fade. A doctor's opinion needs refreshing.
- \\ \\ Anecdotal: steep drop. A Reddit post from 3 months ago is noise.

The math is simple: $\text{confidence}(t) = \text{initial} \times 0.5^{(t / \text{half_life})}$. Tier 0 sources have no half-life—they're permanent until superseded by a newer regulatory action.

3. Time Travel

"What was the known risk profile when I started this medication in June 2023?"

Episteme preserves every historical state. You can query what was believed at any point in time. This matters for:

- **Liability:** "What did we know when we made that decision?"
- **Learning:** "How did our understanding evolve?"
- **Audit:** "Why did the AI recommend that?"

4. Different Questions, Different Answers

The same data can be queried with different **Lenses**:

Lens	Question	Answer Style
Consensus	“What do most sources agree on?”	The most common answer
Authority	“What do trusted sources say?”	Weighted by source tier
Recency	“What’s the latest?”	Most recent claim wins
Skeptic	“Where is there disagreement?”	Shows all claims with conflict scores
Layered	“What does each tier believe?”	Tier-by-tier breakdown

The **Skeptic** lens is particularly powerful: instead of hiding disagreement, it surfaces it. “Here’s where clinical trials and patient reports diverge.”

Use Cases

Consumer Health Intelligence

The Living Review: A continuously updated assessment of a drug or treatment that:

- Shows regulatory, clinical, and patient evidence separately
- Surfaces emerging signals from patient communities before clinical confirmation
- Time-travels to “what was known when you started treatment”
- Alerts returning users to changes since their last visit

Value: Patients and doctors see the full picture, not a false consensus.

Financial Due Diligence

The Contradiction Detector: Multiple analyst teams assess a target. The system:

- Holds all revenue/liability estimates without forcing resolution
- Shows where teams agree (high confidence) vs. disagree (investigate further)
- Tracks which sources informed which conclusions
- Cascades updates when a source is retracted

Value: Acquirers see the variance, not a false precision. \$180M mistakes become visible.

DevOps & Production Safety

The Config Guardian: AI agents deploy infrastructure changes. The system:

- Holds specs from RFCs, internal wikis, runbooks, and Stack Overflow with source tiers
- Blocks deployments when high-tier sources (RFCs, approved runbooks) conflict with the agent’s chosen config
- Auto-escalates to human review when conflict score exceeds threshold
- Provides postmortem audit: “The agent chose X because it read Y, but Z contradicted it”

Value: Agents can’t silently ignore authoritative specs. Conflicts surface before they become outages.

AI Agent Collaboration

The Shared Memory: Multiple AI research agents explore a topic. The system:

- Lets each agent contribute observations with confidence scores
- Resolves conflicts based on agent reputation (trust scores)
- Maintains audit trail: “Agent A believed X because it read Y”
- Supports pre-flight checks: “Must use regulatory sources for this query”

Value: Agents can disagree productively. Reasoning is auditable.

The Product: Augmented Browsing

Episteme is the core. The first application built on it is a browser extension that turns the entire web into an epistemic workspace.

Phase 1: The Benign Layer

A read-only experience. You highlight a claim, the extension looks it up in Episteme and shows you what it knows — sources, tiers, conflicts, history. A lookup tool. No data leaves the browser unless you ask.

This phase builds trust. It demonstrates the value of seeing the full picture on a claim you already care about.

Phase 2: The Active Layer

The extension wakes up. As you browse, it identifies claims on every page — health claims in an article, revenue projections in a pitch deck, technical assertions in a blog post, product promises on a landing page.

For every claim it finds, it **votes**: contributing a signal back to Episteme. What was claimed. Where. When. By whom. That vote is lightweight — a structured observation, not an opinion.

In return for that vote, the extension **overlays everything**:

```
"Semaglutide has no serious GI side effects"

Source: HealthBlog.com (Tier 5 · Anecdotal)
Conflict Score: ██████████ 0.82

▼ Competing claims (4 sources)
  FDA Label (Tier 0): gastroparesis warning added
  NEJM Trial (Tier 1): no signal in Phase III
  Patient Registry (Tier 4): 340 reports
  This page (Tier 5): "no serious side effects"

▼ Decay: this claim is 8mo old, confidence 0.11
▼ Timeline: 3 major shifts since publication
```

Every detail. Source tier. Decay status. Conflict score. Who agrees, who disagrees, and how that's changed over time. The context that was always there, hidden behind the false certainty of a single source.

The Exchange

You give signal. You get transparency.

Every page you visit gets richer — not because the extension injects opinions, but because it surfaces the epistemic structure underneath every claim. The more people participate, the more claims get voted on, the more complete the overlay becomes.

This isn't a fact-checker. Fact-checkers pick a side. This shows you **all the sides**, weighted by who said them and how much that matters.

The Key Insight

Traditional databases optimize for consensus. They want one answer.

Episteme optimizes for epistemic honesty. It wants you to see:

- What different sources believe
- How confident they are
- Where they disagree
- How beliefs have changed over time

The right answer depends on context. A regulator needs the FDA label. A patient needs to know what other patients experienced. A researcher needs to see the emerging signal that hasn't reached the FDA yet.

One database. Different questions. Honest answers.

Technical Foundation (For the Curious)

Episteme is built on a few core principles:

1. **Append-Only:** Claims are never deleted or modified. New claims supersede old ones, but history is preserved.
2. **Content-Addressed:** Every claim has a unique ID based on its content (like Git commits). Same content = same ID.
3. **Cryptographically Signed:** Every claim is signed by the agent that made it. You can verify who said what.
4. **Merkle DAG:** Claims form a directed graph. When a source is retracted, all downstream claims can be identified instantly.

The result: a database that acts more like a **version control system for knowledge** than a traditional data store.

When Is Episteme the Right Choice?

Scenario	Episteme?	Why
Multiple sources report different things	Yes	Core use case
You need to weight sources by authority	Yes	Source class hierarchy
You need to surface disagreement	Yes	Skeptic lens
You need historical snapshots	Yes	Time-travel queries
You need audit trails	Yes	Query audit + signatures
You have one source of truth	No	Use Postgres
Data never conflicts	No	Use Postgres
Consensus is pre-determined	No	Use Postgres

The Bottom Line

Episteme doesn't tell you what's true. It shows you what different sources believe and lets you decide.

For domains where truth is contested, evolving, or depends on perspective—health, finance, research, intelligence—that's exactly what you need.

The alternative is a database that forces false certainty. And false certainty has real costs.