

RELIABILISM FOR THE AGENTIC ERA

Provenance and justified deference in shared human–agent memory

Powerhouse Research, Knowledge Vault team

Powerhouse · apeiron@powerhouse.inc

github.com/apeiron-M/bai-knowledge-note · knowledgevault.powerhouse.io/methodology

ABSTRACT

Teams now share one memory with their AI agents. Agents read it to act and write to it when they finish. The memory grows faster than any person can read it, and most of what it holds was never checked by a person. We argue that this puts a classical question of epistemology, what separates knowledge from a lucky guess, at the centre of software design. Checking outputs cannot answer it. A correct answer from an unreliable process is a lucky guess, and a reader who trusts a usually reliable process that was right this time by luck is in a Gettier case. Agent systems produce both at scale. Process reliabilism (Goldman 1979) gives the right standard: a claim is worth what the process that produced it is worth. The standard has been hard to apply because nothing in ordinary life records which process produced a belief (Conee and Feldman 1998). We show that event-sourced document models with signed operation logs record exactly that, at the moment of production: the author, the mechanism, the source, the reviewer and the time. This moves the generality problem from metaphysics to an engineering decision about what to record. We read the recent literature on agent memory, context degradation, memory contamination and chain-of-thought faithfulness and find that the failures it reports are failures of provenance before they are failures of truth. We then turn from the claim to its reader. Nobody can check every claim in a memory shared with agents, and checking claims outside one's own competence is itself an epistemic failure, a form of epistemic trespassing (Ballantyne 2019). The rational alternative is deference, and recent work on justified algorithmic deference defines justified deference to AI as reliance grounded in higher-order evidence about the reliability, accountability and domain of the source, and asks whether such deference is possible at all (Todorović et al. 2026). Generative AI lacks the institutions that supply such evidence for human experts. We argue that a provenance-keeping memory can supply them for the claims it holds: a named author, a source, a review by a distinct and competent person, a visible disagreement, a correction that propagates, and a track record per process. On this view the object of rational deference is not the model but the record, and a knowledge vault is a deference tool. We state eight invariants for shared human–agent memory, describe their implementation in Knowledge Vault and the Powerhouse platform around it, and mark what remains open. The claim is structural and modest. A signed, replayable record of process makes a shared memory ready for reliabilist evaluation. Reviews by people, kept in the same record, complete it. Together they give a reader what she needs to defer with justification rather than verify without competence or accept without grounds.

Keywords: reliabilism, provenance, epistemic deference, higher-order evidence, epistemic trespassing, epistemic corruption, agent memory, event sourcing, human oversight, testimony, generality problem

1 INTRODUCTION

In a knowledge-rich world, progress does not lie in the direction of reading and writing information faster or storing more of it. Progress lies in the

direction of extracting and exploiting the patterns of the world so that far less information needs to be read, written, or stored.

Herbert A. Simon (1971, p. 46)

Three facts describe the agentic workplace.

First, agents produce more than people can read. The length of software task that frontier agents complete with 50 percent reliability, measured in the time a skilled person needs, has doubled about every seven months since 2019 (Kwa et al. 2025). Each finished task leaves notes, summaries, files and decisions. A team of ten with a dozen agents receives more written output in a week than it could review in a month.

Second, this output goes into shared memory. Agent memory has moved from the context window to persistent stores (Packer et al. 2023; Xu et al. 2025; Chhikara et al. 2025), and from one agent to fleets that share a store (Margalit et al. 2026). Open protocols make it routine for an agent to read a company's documents and write back into them: the Model Context Protocol connects agents to data sources (Anthropic 2024), and the Agent Skills specification packages procedures that agents load and run (Anthropic 2025a). The memory a team relies on is now co-written by software.

Third, people accept what they find there. Automation bias is well documented: under load, people take machine output as settled (Parasuraman and Manzey 2010). Knowledge workers who trust generative AI report less critical thinking when they use it (Lee et al. 2025). Across 106 experiments, combinations of a person and an AI system performed on average worse than the better of the two alone, with losses concentrated in decision tasks (Vaccaro et al. 2024). AI systems can distort what people believe, and the distortion persists after the system is gone (Kidd and Birhane 2023).

More output means less trust. Trust, in Luhmann's sense, is how a person acts despite complexity she cannot inspect (Luhmann 1979). When the volume of claims outruns the capacity to check them, a team has two options. It can trust everything, which is credulity. Or it can trust what carries a mark of having been checked, which requires that the mark exist and that it be reliable.

The usual advice, to think critically and verify for oneself, is a third option only in appearance. The cost of verification escalates when generation is nearly free (Ferrara 2026), and outside a person's own domain it is not a virtue but a failure: the verifier passes judgement where she lacks the competence to judge (Ballantyne 2019; Levy 2022). Social epistemology has long held that for most of what we know, deference to a well-placed source is more reliable than our own first-order evaluation (Hardwig 1985; Zagzebski 2012; Matheson 2022). Todorović et al. (2026) frame the resulting dilemma for generative AI sharply. Verifying its outputs is untenable and leads to epistemic trespassing; deferring to them blindly leads to algorithmic heteronomy, the loss of rational autonomy to an opaque system. The way between is *justified algorithmic deference*: reliance grounded in higher-order evidence about the source's reliability, its institutional context and its domain of competence. The second option above, trust in what carries a mark, is that way, provided the marks are the right higher-order evidence. Much of this paper is about what makes them so.

We call what is at stake *epistemic corruption*. Kidd (2019) uses the term for conditions that damage a person's epistemic character. We use it for a shared memory: a store is corrupted when its contents can no longer be told apart by origin, warrant or review, so that a reader, human or agent, cannot tell a checked claim from a guess. Corruption in this sense needs no false claims. A memory can be corrupted while every sentence in it is true, because nobody can tell which sentences those are.

Our thesis has four parts.

1. Process reliabilism is the right standard for shared human-agent memory. The question to ask of a claim is not only whether it is true but whether the process that produced it is one that produces truths.
2. The instrument reliabilism lacked is an operation log. An event-sourced record of typed, signed operations captures the process identity of each claim at the moment of production, which is what the generality problem said could not be had.
3. Provenance, understood as a resolvable record of who asserted what, from what source, by what mechanism, checked by whom, is the condition under which a claim in shared memory can count as knowledge for the people who depend on it. Without it, correct output is luck.
4. The same record is what makes deference to the memory rational. It carries the higher-order evidence that a reader without domain competence can assess: who stands behind a claim, whether a competent person other than its author reviewed it, whether it is contested, whether its source still stands, and how often claims of its kind have survived review. A knowledge vault built this way is a deference tool.

It moves the object of deference from the model, which lacks the accountability that deference requires, to the record, which can be built to have it.

Section 2 reviews what goes wrong in shared memory, with evidence from the current literature. Section 3 explains why checking outputs cannot fix it and why reliabilism can, and why asking every reader to check is a form of trespass. Section 4 presents the operation log as the missing instrument. Section 5 turns from the claim to its reader and argues that the record is what makes deference justified. Section 6 states eight invariants. Section 7 describes their implementation in Knowledge Vault and the platform around it. Section 8 answers objections. Section 9 concludes.

2 WHAT GOES WRONG IN SHARED MEMORY

2.1 Context grows; attention does not

Simon (1971, p. 40) observed that "a wealth of information creates a poverty of attention". Agent systems have made this an engineering constraint. Language models do not use their context uniformly. Relevant material placed in the middle of a long context is retrieved worse than material at either end (Liu et al. 2024). Across eighteen models, performance on simple tasks fell as input grew, well before the context window was full (Hong et al. 2025). Practitioners now treat context as a scarce resource to be engineered (Anthropic 2025b).

The consequence for shared memory is direct. Each task pulls more notes, files and past sessions into the agent's context. Nothing in a full context window marks which part a person checked, which part is a guess and which part is out of date. The agent weighs them the same. So does the person who reads the result.

Simon drew the design conclusion in the same essay, and it stands at the head of this paper: progress lies in extracting the patterns of the world so that far less needs to be read, written or stored. For a memory shared with agents, the pattern that saves the most reading is provenance. A reader who can see who wrote a claim, from what, and whether anyone checked it can leave most of the store unread and still act on it. A reader who cannot must read everything, or trust everything.

2.2 Reasoning does not survive a handoff

Two lines of work address what a model's reasoning reveals. Chain-of-thought traces give a partial view of how a model reached an answer, and they are unfaithful often: in one study, models that used a hint to reach an answer mentioned the hint in their reasoning 25 percent of the time for one model and 39 percent for another (Chen et al. 2025). Korbak et al. (2025) call the ability to monitor reasoning traces a new and fragile opportunity. Shah and Dragan (2026) argue that the window is closing as architectures change, and ask developers who drop transparent architectures to show that their replacements stay as monitorable.

Both properties belong to one model inside one turn. Shared memory starts when the turn ends. Agent A writes a note. Agent B, or a person, reads it a month later. The chain of thought is gone. The confidence number, if there is one, is a self-report with no recorded basis. The reader sees a claim and nothing else. Whatever monitorability the model had, it did not travel.

2.3 Three contaminations

When agents write into a memory that people treat as their own thinking, three things degrade. Each has a basis in cognitive psychology, and each has now been observed in agent systems.

Authorship erosion. People remember content longer than they remember its source. This is the source monitoring problem (Johnson, Hashtroudi and Lindsay 1993): the judgement of where a memory came from is a reconstruction, and it fails before the memory does. In a shared store, a few months later, nobody can tell what a person wrote from what an agent drafted. The memory drifts from *what we know* to *what some process once asserted*.

Confidence laundering. Gilbert (1991) argued that comprehension involves acceptance by default; rejecting a claim takes a second, effortful step that fails under load. An agent extracts a claim with low confidence. The claim enters memory as a flat statement. A person, under load, accepts it. Later she cites it as settled. The hedge was lost at the first write.

Reflexive contamination. Agents read the memory to inform their next output. Once agent-authored material is in it, the agent reads its own prior assertions as the team's settled knowledge. The loop closes and

the signal degrades. The pattern has analogues at every scale. Models trained on their own outputs collapse toward a narrow distribution (Shumailov et al. 2024). Social influence between people destroys the independence that makes group estimates accurate (Lorenz et al. 2011). Echo chambers do their damage by discrediting outside sources, which a memory with no source markers does by default (Nguyen 2020). In multi-user agents, artifacts left by one user’s benign interaction degraded other users’ outcomes at rates of 57 to 71 percent, with no attacker involved (Yang et al. 2026). A study of fleet memory names four failure modes, among them stale propagation, contradiction persistence and provenance collapse (Margalit et al. 2026). A survey of persistent-state agents finds that the literature concentrates on accumulating and retrieving state and neglects governing, recovering and relinquishing it (Ding et al. 2026). Memory systems that filter retrieval by recorded ancestry and permission, rather than by semantic similarity alone, are a first response (Wang et al. 2026).

2.4 Adversaries need very little

Where honest use contaminates, hostile use does worse at lower cost. Poisoning less than 0.1 percent of an agent’s memory or knowledge base produced attack success rates above 80 percent, for queries carrying an optimised trigger, with under one percent loss in benign performance (Chen et al. 2024). Injecting a handful of texts into a retrieval corpus steered answers to attacker-chosen targets (Zou et al. 2025). Indirect prompt injection turns any document an agent reads into a channel for instructions (Greshake et al. 2023). In each case the poisoned entry is indistinguishable, at read time, from an honest one. Hammond et al. (2025) list such failures among the risks specific to systems of many agents. An empirical taxonomy of multi-agent failures sorts them into failures of system design and specification, inter-agent misalignment, and task verification (Cemri et al. 2025).

2.5 The pattern

In each case the memory fails as a record before it fails as a body of truths. The context holds checked and unchecked claims without distinction. The reasoning that produced a claim is gone by the time anyone reads it. Authorship, confidence and independence erode. A poisoned entry looks like an honest one. These are failures of provenance. Truth, where it fails, fails downstream.

3 WHY CHECKING OUTPUTS IS NOT ENOUGH

3.1 Gettier at scale

Gettier (1963) showed that a belief can be justified and true and still fall short of knowledge, when the justification and the truth are connected by luck. Russell’s (1948) stopped clock is the stock illustration. A person reads the right time from a clock that stopped twelve hours earlier. Her belief is true, and justified, because she has good reason to trust the clock, but she does not know the time. Shared memory creates this structure at scale. A reader who trusts a pipeline that is usually right, and that this time was right by luck, is in her position. Where the pipeline is not reliable at all, the output is not even justified; it is a true guess. Either way the output passes every output check.

Agent systems invite such cases. Training and evaluation reward a confident guess over an admission of uncertainty, so models guess (Kalai et al. 2025). Benchmarks score outputs, and agents are tuned to them (Kapoor et al. 2024). Output scoring cannot distinguish a reliable process from a lucky one, because it never looks at the process. A memory filled by output-scored agents is a memory of claims whose connection to truth is unknown.

3.2 Reliabilism

Goldman (1979) proposed that a belief is justified when a reliable process produced it: a process that tends to produce true beliefs across the range of cases it meets. Reliabilism changes the object of evaluation from the belief to the process. It is a theory of justification, not by itself an answer to Gettier: a reliable process can still deliver a true belief by luck, which is why accounts of knowledge add an anti-luck or safety condition (Pritchard 2005). Virtue reliabilism refines this: knowledge is a success that is creditable to the knower’s ability (Greco 2010), and an ability is a way of forming beliefs that has been tuned to succeed (Kelp 2019).

Reliabilism is the natural standard for agent output because it is the only standard that asks the question an engineer can act on. *Was this answer right* tells you about one answer. *Does this process produce right answers* tells you what to keep, fix or remove. Durán and Formanek (2018) make the same move for computer simulations under the name computational reliabilism: trust in an opaque computation is warranted by indicators of its reliability, among them a history of successful and unsuccessful runs. Alvarado (2023) argues that AI systems are epistemic technologies, designed to produce and operate on knowledge, and must be assessed as such.

3.3 The generality problem, and why agents sharpen it

Reliabilism has a known difficulty. Every belief is produced by a process that falls under indefinitely many types. Which type's reliability counts? Conee and Feldman (1998) argue that nothing in the epistemic situation picks one out, and that reliabilism is empty until it does.

For agents the problem is sharper and, we will argue, more tractable. The candidate types for one extracted claim include: extraction by a language model; by model family M ; by version $M.n$ with prompt revision p ; using skill s at content hash h ; from a source of genre g ; at chunk index k with the context c percent full. Reliability does vary across these. Section 2.1 showed it varies with context fill. The relevant type is finer than any a person would name, and in ordinary systems it is not recorded. A file in a shared drive carries its content and nothing about the process that produced it.

Kelp (2019) defuses the problem by indexing reliability to an agent using a way of belief formation in a range of conditions, where the way must be *grounded*: tuned by learning or selection toward success in those conditions. Apply this to an extraction pipeline and the ability decomposes. Trained weights are tuned, but toward next-token prediction and preference, adjacent targets to extracting true claims from a document. A written procedure is tuned by its designer toward the success she had in view. Enforcing checks secure well-formedness, not truth. The human operator is the one bearer who can come to know that the pipeline works, and her knowledge grounds *her* reliance on the system, not the system's own ability. No single bearer is grounded throughout. The pipeline is a distributed ability with no competent agent.

This result explains a common confusion. The question *is the agent reliable* has no answer because it presupposes a bearer the architecture does not supply. The answerable question is: *is this process type, in these conditions, reliable*. That question needs records.

3.4 Self-assessment is not assessment

A tempting fix is to have the agent check itself. The evidence is against it. Large language models do not reliably correct their own reasoning without external feedback (Huang et al. 2024). When a model judges outputs, it recognises and prefers its own (Panickssery, Bowman and Feng 2024). Models produce text with indifference to its truth where the training signal does not reward truth (Hicks, Humphries and Slater 2024).

Reliabilism gives the reason. A check run by the same process type as the production adds little independent evidence, because its errors are correlated with those it is meant to catch (Section 5.4). Work on AI control reaches the same design from the safety side: use a different, trusted process to monitor an untrusted one, and keep a person in the audit loop (Greenblatt et al. 2024); work on scalable oversight makes a related case for supervising models with weaker but trusted checkers (Bowman et al. 2022).

There is a second, older point. BonJour (1980) imagined Norman, who has a reliable clairvoyant faculty but no reason to think so. Most readers judge Norman unjustified. The lesson for shared memory is that reliability the reader cannot see does her no good. A team that relies on agents needs access to the record that shows the process was reliable, not an assurance that it was.

3.5 Checking is trespassing

The previous subsections argued that an output check cannot tell a reliable process from a lucky one. There is a second limit, on the side of the person doing the checking. Most readers of a shared memory are not competent to check most of what is in it.

Ballantyne (2019) calls it *epistemic trespassing* when a person competent in one domain passes judgement on questions in another, where she lacks the evidence or skills to answer well. Asking every reader to verify agent output asks her to trespass whenever the claim falls outside her field, which in a team's shared memory is most of the time. The general skills of critical thinking do not rescue her. They transfer poorly across domains (Willingham 2008), and even experts' skills are brittle outside the conditions in which they were

acquired (Kilov 2021). Levy and Varley (2026) add that what novices lack is not only knowledge but the *guardrails* that experts learn from long practice, the sense of when a question exceeds their competence and when to defer. Generated text removes the cues that would trigger those guardrails. It is fluent, well formatted and confident in every domain, and it offers rationalisations that readers mistake for faithful explanations of how it reached its answer (Jacovi and Goldberg 2021). Verification also fails on cost. When generation is nearly free, the cost of verifying what is generated escalates until it becomes prohibitive at scale (Ferrara 2026).

Todorović and Bregant (manuscript) make the argument for climate communication: a layperson applying first-order critical thinking to a sophisticated sceptical argument can assess its structure but not its premises, and is led astray exactly where it matters. Todorović et al. (2026) extend it to generative AI and draw the conclusion we adopt. Critical thinking should be redirected, not abandoned: from first-order evaluation of claims, which belongs to those competent in the domain, to the evaluation of higher-order evidence about the source, which a layperson can do (Anderson 2011).

For shared memory this has a design consequence that reliabilism alone does not supply. The review that makes a claim canonical must be done by someone competent to do it, and everyone else must be able to see that it was. The first is a constraint on who reviews. The second is a constraint on what the record shows the reader. Section 5 takes up both.

4 THE OPERATION LOG AS RELIABILISM'S INSTRUMENT

4.1 From files to operations

Event sourcing stores a system's state as an append-only sequence of the changes that produced it; current state is a replay (Fowler 2005). Immutable logs are cheap, and they turn history into data (Helland 2015). Local-first software adds that the data lives with its owner and syncs rather than being held on someone else's server (Kleppmann et al. 2019).

A *document model* applies this to a document. It has a state schema, an action schema and a reducer: a pure function from state and action to new state. Actions are expressions of intent. The runtime applies them through the reducer and appends an operation: the action, its ordinal, its timestamp, the hash of the state before and after. In the Powerhouse reactor, each operation also carries the signer's key and the application that made the change, and the signature covers the timestamp, the signer identity, the action hash and the previous state hash, so that an operation cannot be moved to another point in history without detection (Powerhouse 2026). Figure 1 shows the stack.

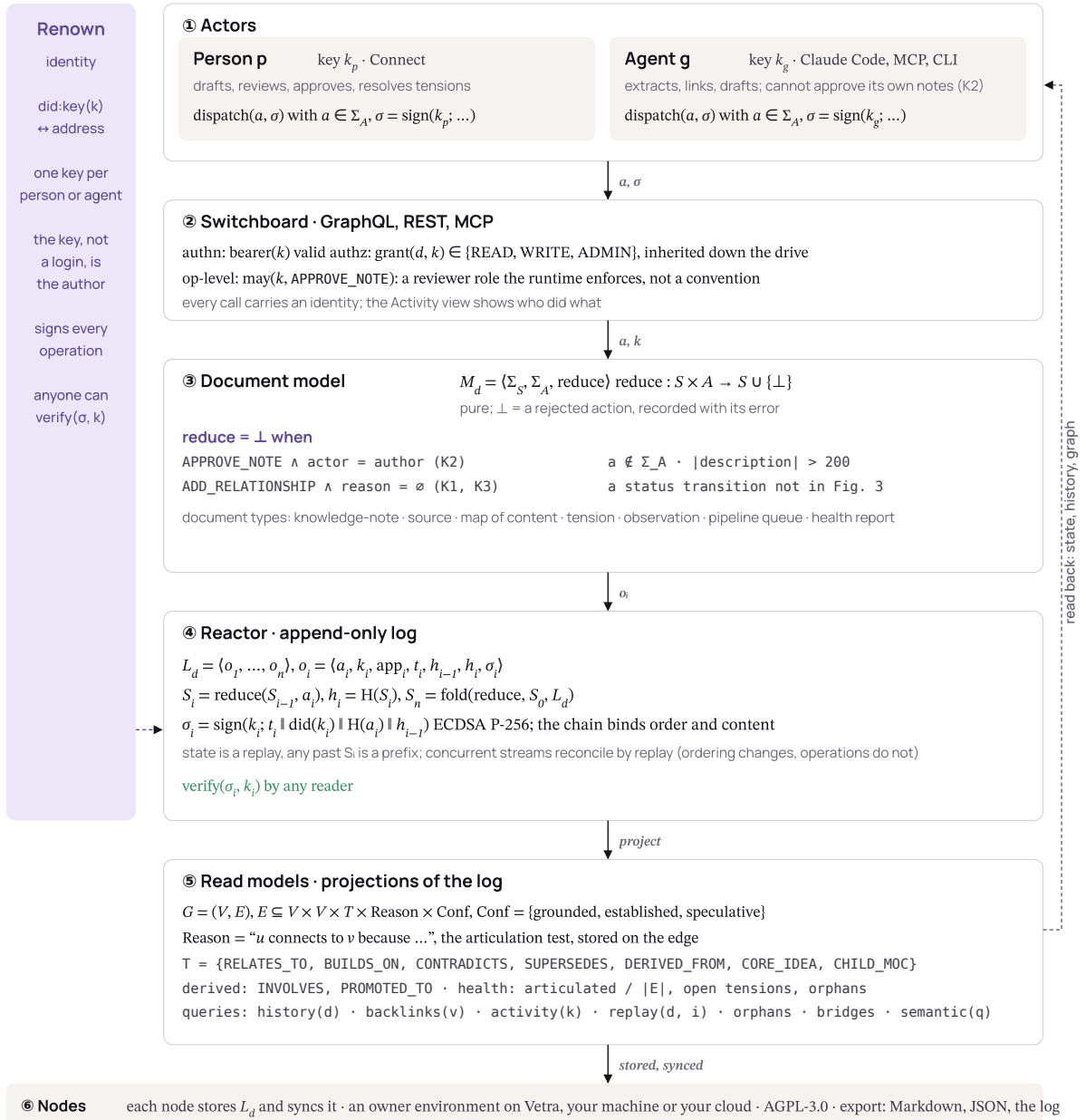


Figure 1: Data flow through the Powerhouse stack. A signed action from a person’s or an agent’s key passes authentication and authorisation at Switchboard, is validated by the document model’s pure reducer (which returns $\perp$ for a rejected action and records the attempt), and is appended to the document’s log as an operation $\langle a, k, \text{app}, t, h(i-1), h(i), \sigma \rangle$. State is a fold of the log; the graph and every query are projections of it; each node stores and syncs the log.

A skill invocation, in this architecture, is not a tool call that leaves a log line. It is an action against a document, dispatched by an attributed actor, executed by a deterministic reducer, and recorded as an operation. The document carries its own history. Every byte of the final state traces to an operation, which traces to an actor and a mechanism.

4.2 What the log records is the process-type key

Section 3.3 asked which process type’s reliability counts and observed that the relevant parameters are not recorded. In an operation log they can be. Per claim, the log can carry the model identity and version, the prompt hash, the skill name, version and content hash, the chunk index, the context-fill position, the source reference and span, the signer, and the time. This is the process-type key the generality problem asks for, written down at the moment of production rather than reconstructed later.

The problem is not solved. It is relocated, from metaphysics to key selection. A key too fine has no outcomes per cell and predicts nothing. A key too coarse hides the variation that matters. The right granularity is a bias-variance question, and it is one the system can answer about itself once it has outcomes to count. That is a tractable empirical question where there was an impasse.

4.3 Provenance in the database sense

The database community has studied provenance for two decades. Buneman, Khanna and Tan (2001) distinguish *why*-provenance, the source data that justifies an output, from *where*-provenance, the location the output was copied from. The W3C PROV model records entities, the activities that produced them and the agents responsible (Moreau and Missier 2013). Kroll (2021) argues that traceability, the ability to reconstruct how a system worked, how it was built and why, is the principle that makes accountability operational in computing systems. Chan et al. (2024) propose agent identifiers, real-time monitoring and activity logs as the three measures that give visibility into AI agents.

A knowledge edge in the system we describe does the *why*-provenance work in a form a person can read. The edge from a note to its source carries a *reason*, a sentence that says what in the source warrants the claim, and a *confidence* from a closed vocabulary. We call the requirement that the sentence exist the articulation test: a link that cannot complete *A connects to B because* is not knowledge, it is an address-book entry. The health report counts articulated edges against all edges, and the ratio is a metric the team watches.

4.4 Ready versus in action

A log records what was produced. It does not record whether it was true. Reliability is a property of a process type measured against outcomes. Without an outcome channel, the log is provenance: attribution, and attribution is not reliability.

The outcome channel exists when reviews are kept in the same log as the claims they review: approvals and rejections by a named reviewer, resolutions of contradictions, retractions of sources. Each is a signed operation keyed to the same process type as the claim it judges. Then the reliability of each key is a query. The history-of-runs indicator that Durán and Formanek (2018) name becomes a column.

Hence the slogan we use. Document models make a system *reliabilism-ready*. Human verdicts recorded against the process key make it *reliabilism in action*. The second does not come free with the first, and a system that has the first and claims the second is laundering.

4.5 Testimony, vigilance and the division of epistemic labour

Should agent output be treated as testimony? Philosophers disagree. Wheeler (2020) argues that some robots can testify because they can be objects of epistemic trust. Freiman (2024) proposes a concept of AI-testimony for outputs that are propositional, delivered with no person in the loop and received as if from a speaker. Lackey (2008) locates the reliability that matters for testimony in the statement rather than the speaker's belief, which lets a source with no beliefs still be assessed. We take no side on whether agents testify. We take the practical point that a reader treats their output as she treats testimony: she accepts it on trust she cannot ground in her own inspection.

Hardwig (1985) showed that such dependence is rational when the trust is grounded, and that most of what any person knows rests on it. Craig (1990) argued that the concept of knowledge exists to flag good informants: to mark whom to ask. A shared memory that holds claims from many informants, human and machine, needs the flag to be visible on the claim. Sperber et al. (2010) describe the mechanisms of epistemic vigilance by which people calibrate trust in communicated information, toward the source and toward the content. Vigilance needs cues. A claim with no author, no source and no review mark gives none, and under load the Gilbert default, acceptance, wins. Goldberg (2010) argues that a hearer's justification depends on the speaker's cognitive processes, so that reliabilism extends across persons. It extends across agents and logs by the same argument. The log supplies the cues that vigilance needs and that the context window strips away.

5 DEFERENCE BY DESIGN

Sections 3 and 4 concerned the claim: what makes it knowledge, and what record makes that assessable. This section concerns the reader. Given that she cannot check most claims, and should not try, under what conditions is it rational for her to act on what the memory says?

5.1 Three relationships to a source

Todorović et al. (2026) distinguish three epistemic relationships a person can have to generative AI. In *justified deference*, she relies on its output for reasons she can state, grounded in higher-order evidence about the system's reliability, its institutional context and its domain of competence. In *epistemic trespassing*, she tries to verify the output herself in a domain where she lacks the competence (Section 3.5). In *algorithmic heteronomy*, she relies on the output without any rational basis, and her judgement is gradually replaced rather than extended. They define justified algorithmic deference as an agent's *ex ante* rational reliance on generative AI output, grounded in her assessment of such higher-order evidence, and they leave open, as the central question of their project, whether a deference of this kind is possible toward systems that lack the accountability of human experts.

The third relationship is the one the agentic workplace slides into by default. Lange (2026) warns against *AI preemptionism*, the view that a system's output should replace the user's own reasons rather than join them, and proposes instead a *total evidence* view on which the output is one contributory reason, weighed together with higher-order evidence about the system's reliability and domain. Coeckelbergh (2026) argues that generative AI acts as an epistemic filter that revises belief in ways that reduce epistemic agency. Frischmann and Selinger (2018) describe technologies that, by doing our thinking, re-engineer the people who use them. Each of these is the condition this paper calls corruption, seen from the reader's side.

Deference is not the problem. Hardwig (1985) showed that epistemic dependence is a structural feature of modern knowledge, Zagzebski (2012) that consistent self-reliance is incoherent, and Carter (2020) that intellectual autonomy is compatible with, and in a complex world requires, wise dependence. Matheson (2022) and McKenna (2023) argue that we often do better, by the standard of getting things right, not to think for ourselves. The problem is deference without grounds.

5.2 What a reader can assess

Christensen (2010) distinguishes first-order evidence, which bears on whether a claim is true, from higher-order evidence, which bears on whether one's assessment of it, or a source's, is reliable. Goldman (2001) asked how a novice can choose between experts she cannot evaluate directly and listed the sources of evidence open to her: the experts' arguments, agreement with other experts, appraisals by meta-experts, evidence of interests and biases, and track records. Anderson (2011) argued that laypeople can assess testimony through such indicators without domain expertise: whether the source is an expert, whether it is honest, whether it is epistemically responsible, and where the consensus lies. Origgi (2012) showed how reputation works as a proxy for reliability, and Wilholt (2013) how trust in science rests on shared conventions of the research community rather than on individual inspection.

Each of these indicators has a counterpart that a shared memory can record and display on the claim itself. Table 1 sets them side by side.

	Higher-order indicator	What the record shows on a claim
K1, K3	Who asserts it; whether that source is an expert (Goldman 2001; Anderson 2011)	Signed author, human or agent, with the application and model that produced it
K1, K3	What it rests on (Goldman 2001: arguments)	Source edge with a reason sentence and a confidence from a closed vocabulary, or an honest UNKNOWN
K2	Appraisal by others (Goldman 2001: meta-experts; Anderson 2011: epistemic responsibility)	Review state, and the signature of a reviewer distinct from the author
K6	Agreement and disagreement (Goldman 2001; Oreskes 2021)	Open tensions shown on both contested claims and on their maps; agreement counted by independent lineage, not by signer (Section 5.4)
K5	Whether the ground still stands	Retraction of a source marks every claim derived from it
K8	Track record and reputation (Goldman 2001; Origgi 2012)	Share of claims from the same process key that survived review

Table 1. Higher-order evidence a reader without domain competence can assess, and where a provenance-keeping memory records it. In Knowledge Vault today the K5 and K8 rows are not yet in place: retraction does

not propagate, and track records per process key are not computed (Section 7). Agreement is counted by signer, not yet by independent lineage.

None of these requires the reader to evaluate the claim's content. Each is legible to someone who knows nothing about the domain, which is what makes deference on its basis available to her rather than only to an expert. The table is also a test of a design: an indicator that the record cannot show is one the reader must take on trust, and trust of that kind is the heteronomy of Section 5.1.

5.3 The accountability gap, and where to defer

The strongest objection to deferring to AI is a disanalogy. Rational deference to a human expert is grounded in institutional accountability: peer review, sanctions for dishonesty, reputation and shared norms. Trust in science, on Oreskes's (2021) account, is warranted by the social process that vets its claims: critical scrutiny by a diverse community with means of correcting its errors (see also Longino 1990). As Todorović et al. (2026, §23.1.2) put it, "the disanalogy is not between hard and easy claims but between accountable and unaccountable sources", and a language model has no epistemic community that holds it to account.

We accept the disanalogy for the model. Our proposal is to change what is deferred to. A reader of a shared memory does not need to defer to the model that drafted a claim. She can defer to the record of what happened to the claim afterwards, and that record can be built to have the features the model lacks. Debate has a venue: a contradiction opens a tension that stays open until a person settles it (K6, after Longino 1990). Diverse perspectives are enforced at the point of review: the author cannot approve her own claim (K2). Correction is a mechanism rather than a hope: a retracted source reaches every claim that rested on it (K5). Reputation becomes a measured track record per process type, kept in the same log as the claims it judges (K8). And every step is signed, so that dishonesty is attributable rather than anonymous.

This does not make the model an epistemic authority, and it does not answer the general question Todorović et al. (2026) pose about laypeople and public AI systems. It answers a narrower question: what would a store of AI-assisted claims have to record for deference to its contents to be justified? Our answer is the operation log of Section 4 together with the invariants of Section 6. A memory with these properties is a deference tool, in the sense that its purpose is to make rational deference possible for readers who cannot verify, rather than to make verification easier for readers who cannot be expected to do it.

5.4 Agreement, dissent and independence

Agreement is the indicator a reader is most tempted to count. If several sources say the same thing, that seems to settle it. It does so only under a condition. Condorcet's (1785) jury theorem shows that a majority of voters who are each somewhat more likely than not to be right becomes almost certainly right as the group grows, but only if their judgements are independent; when votes are correlated the guarantee weakens or disappears (Ladha 1992; Dietrich and Spiekermann 2013). Goldman (2001) made the same point for experts: agreement adds evidence only if the agreeing parties are not non-discriminating reflectors of a common source, and Lackey (2013) argues that what matters is not the number of believers but the independence of the routes by which they came to believe. Ten newspapers that reprint one wire report are one source.

Multi-agent systems put this condition under strain, as Borut Trpin has pointed out to us (personal communication). Systems such as Google's AI co-scientist organise several agents into something like a research community: some generate hypotheses, others critique and rank them in tournaments of debate, others refine the survivors (Gottweis et al. 2025). Their agreement looks like a consensus in miniature. But the agents often run on the same base model, were trained on overlapping data, and share prompts and context. Their errors are then correlated, and their agreement may carry the evidential weight of a single source while presenting itself as many. Kleinberg and Raghavan (2021) describe the general condition as algorithmic monoculture: when many decision makers rely on the same model, their errors do not cancel but coincide. In a shared memory the same holds for review. A second agent on the same model that approves the first agent's claim is a distinct signer, not an independent check, and so is a person who approves output she prompted herself.

Dissent has the opposite value. Zollman (2010) showed in models of scientific communities that groups which keep a diversity of views for a while reach the truth more reliably than groups that converge quickly, and recent work studies how epistemic and zetetic norms interact when scientists disagree (Justin et al. forthcoming). A recorded disagreement is evidence that at least two routes to a question differ, which is more

than unanimous agreement among copies can show. This is a further reason to keep contested claims visible rather than settle them by overwriting (K6).

Independence has more than one layer: causal (do the sources share an origin, such as a model or a corpus), statistical (do they err together), and of access (did they see the same evidence and context). A signature shows none of them. For agreement to count as higher-order evidence in the sense of Table 1, the record would have to show lineage: which model family and version produced each agreeing claim or review, and what each had in its context. Agreement could then be counted by independent lineages rather than by signers. The operation log records part of this today, the model and skill on each operation, as fields the writer declares; it does not record what the writer read. How to measure independence among agents, and how much correlated agreement should count, are open questions, and in our view among the most important for any system that asks people to trust agreement between machines.

5.5 Guardrails and domain boundaries

A review is higher-order evidence only if the reviewer was competent. K2 requires that the reviewer differ from the author; it does not by itself require that she know the domain. Levy and Varley's (2026) guardrails apply to reviewers as much as to readers, and the metacognitive problem applies too: those who lack competence in a domain are often the least able to tell. Todorović et al. (2026) propose to meet it not by asking people to assess their own competence but by drawing domain boundaries in advance, for instance by specifying domain boundaries such as "any medical, legal, or financial claim" (Todorović et al. 2026, WP4).

A memory can implement this boundary in its access rules. The operation that approves a claim can be restricted to reviewers named for the claim's domain, so that a reviewer outside her competence cannot make a claim canonical, however confident she is. The boundary is then a property of the runtime rather than of the reviewer's self-assessment, in the same way that the reducer, not the reader of a policy, enforces K2. In Knowledge Vault today the approval operation can be restricted to named reviewers for the whole vault; restriction by topic or map is not yet built, and we list it with K2's status in Section 7.

5.6 Deference without heteronomy

A deference tool invites the dependence it is meant to discipline. Todorović et al. (2026), following Murimi (2023), describe a cycle in which reliance on AI reduces autonomy, which increases reliance. Flogie et al. (forthcoming) argue, drawing on Kant's maxims of the common human understanding (Kant 1790, §40), that generative AI undermines all three at once: thinking for oneself, thinking from the standpoint of others, and thinking consistently, and that education should act as a counter-environment to algorithmic governance.

The invariants answer each maxim with a record. Thinking for oneself is preserved at the point that matters, because no agent output becomes canonical without the judgement of a person who is not its author (K2); this is Lange's total evidence view as a lifecycle, in which agent output enters as a draft, one reason among others, and becomes settled only with independent support. Thinking from the standpoint of others is preserved because a disagreement is shown on both sides and is not resolved by overwriting (K6). Thinking consistently is preserved because a withdrawn ground withdraws what it supported (K5). The reader still defers, as she must. But what she defers to is a record of other people's judgement, which she can inspect, contest and add to, not a fluent voice whose reasons she cannot see. That is the difference between extended cognition and heteronomy (Carter 2020; Clark and Chalmers 1998) drawn in an operation log.

6 EIGHT INVARIANTS FOR SHARED HUMAN-AGENT MEMORY

Rules can live in two places. A rule in a prompt or a skill file binds the agent that reads it and weakens as the context fills. A rule in a reducer binds every writer, including an agent that never read the documentation, because the operation that would break the rule does not exist for it to call. Critch (2022) formalises a boundary as a surface whose crossings can be counted; Dalrymple et al. (2024) propose that a safety specification can be stated as a ban on boundary crossings. A reducer is a boundary of this kind. We state

eight invariants that belong in the reducer, each with the failure it prevents and the literature that motivates it.

Invariant	Prevents	Grounds
K1 A derived claim carries a resolvable source reference. Where provenance is missing, the record says UNKNOWN.	False credentials: an assertion of derivation with no referent.	Kroll 2021
K2 Assessment is performed by a signer distinct from the producer.	Self-preference and self-correction failures.	Panickssery et al. 2024; Huang et al. 2024; EU AI Act Art. 14
K3 A confidence value requires a recorded basis. Schema completeness is a separate field.	Confidence laundering.	Lin, Hilton and Evans 2022; Kalai et al. 2025
K4 A coherence drop flags a claim and routes it to a person. It never blocks a write in silence.	Automation that hides its own failures from the operator.	Bainbridge 1983
K5 Defeat propagates. Retracting a source marks every claim derived from it.	Insulation from correction.	Pollock 1987; Doyle 1979; Alchourrón, Gärdenfors and Makinson 1985
K6 Contestation is visible on both sides of an open disagreement.	Contradiction persistence; echo chambers.	Longino 1990; Mill 1859
K7 Every phase handoff carries the fields the next phase is entitled to expect.	Inter-agent misalignment.	Cemri et al. 2025; Torres-Arias et al. 2019
K8 A signed human verdict governs its own claim. It governs the process only once corroborated.	Goodhart effects on review metrics.	Strathern 1997; Power 1997

Three of these deserve comment.

K1 comes from a failure we met in our own vault. A required field with a closed set of values and no honest empty value will be filled with the plausible value. A repair routine asked to fill a missing origin wrote DERIVED with no referent, because the schema offered nothing else. The fix is a schema with an UNKNOWN member and a reducer that refuses a derivation claim without a resolvable target.

K5 is the computational core. Pollock (1987) distinguishes rebutting defeaters, which attack a conclusion, from undercutting defeaters, which attack the link between ground and conclusion. A retracted source is an undercutting defeater for everything derived from it. Doyle’s (1979) truth maintenance system propagated such defeat along recorded justifications, and the AGM postulates (Alchourrón, Gärdenfors and Makinson 1985) describe the rational contraction of a belief set when a ground is withdrawn. Harman (1986) observed that people cannot run a foundations theory of belief revision because they do not keep track of their justifications. An organisation with an operation log can. A broken provenance chain is a chain along which a retraction cannot travel, and that is the main harm of fabricated provenance: the claim it decorates becomes unfalsifiable by the system’s own machinery.

K6 draws on Longino’s (1990) account of objectivity as the product of critical interaction: there must be venues for criticism, uptake of criticism, public standards and shared authority. A recorded contradiction that stays open until a named person settles it is such a venue in miniature. Mill (1859) made the older point that a belief held without exposure to its negation is held as a dead dogma. A memory that resolves contradictions by overwriting loses the record of why one side won.

7 IMPLEMENTATION IN KNOWLEDGE VAULT

Knowledge Vault is a shared memory for teams and their agents, built on the Powerhouse stack: document models for each record type, a reactor that stores and replays signed operations, Switchboard for the query and write interface, Renown for identity, and Connect for the human interface (Powerhouse 2026). We describe how far each invariant is implemented at the time of writing. The vault is open source under AGPL-3.0, so each claim here can be checked against the code.

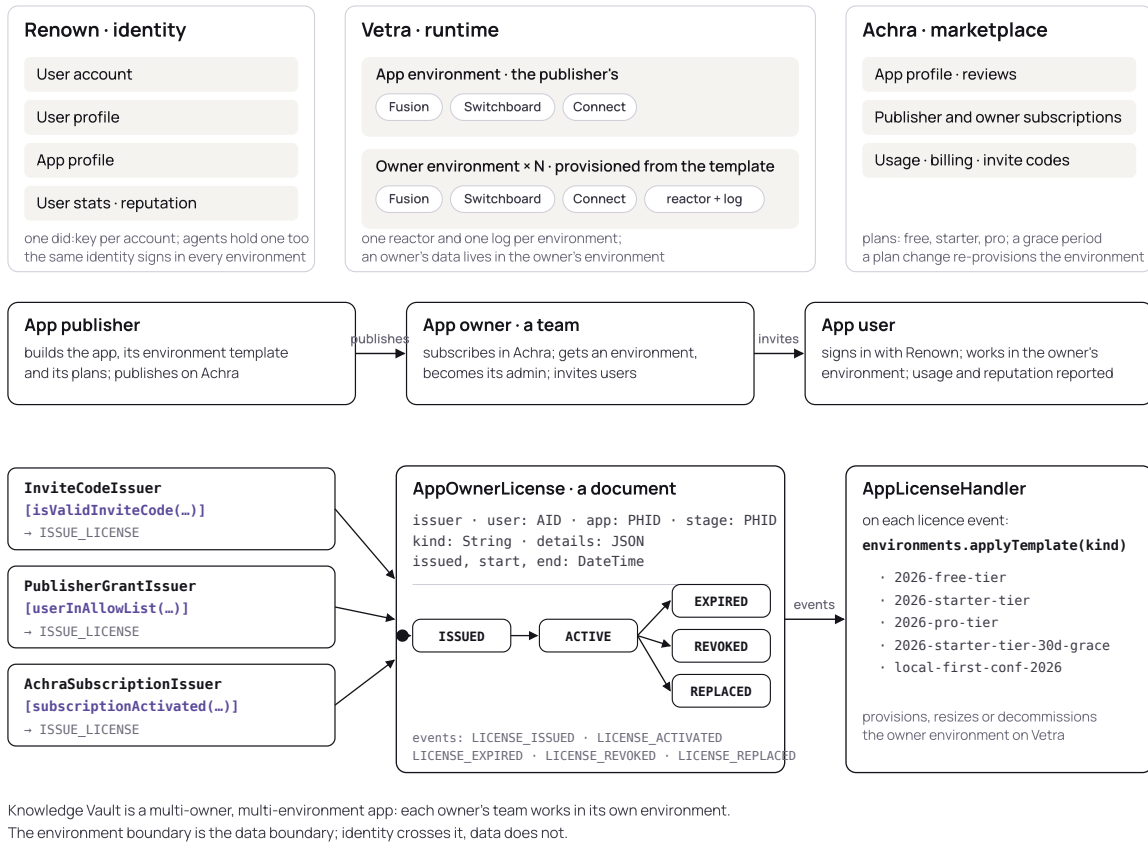


Figure 2: The platform around a vault, as designed. Renown holds identity; Vetra runs one environment per owner from the publisher's template; Achra sells and bills. A licence is a document with a status machine, and a handler maps its kind to an environment template. The environment boundary is the data boundary; identity crosses it, data does not.

The platform around a vault. The platform decides where a team's log lives and who can sign it (Figure 2). Three services take part. *Renown* holds identity: a user account, a profile, an app profile and the user's stats, with one `did:key` per account; the same identity signs in every environment, and agents hold identities of the same kind. *Vetra* runs environments: the publisher's app environment, and one owner environment per team, provisioned from the publisher's template, each with its own reactor, Switchboard and Connect. *Achra* is the marketplace: the app profile and its reviews, publisher and owner subscriptions, usage, billing and invite codes. Three roles follow. The publisher builds the app, its environment template and its plans. An owner subscribes, receives an environment, becomes its administrator and invites users. Users sign in with their Renown identity and work in the owner's environment, which reports usage and reputation back. The right to an environment is itself a document: an `AppOwnerLicense` with an issuer, the owner's identity, the app, a stage, a kind and a validity window. An invite code, a publisher grant or an activated subscription issues it, and it moves through ISSUED and ACTIVE to EXPIRED, REVOKED or REPLACED. A handler maps each licence kind to an environment template, so a plan change provisions or decommissions the environment through the same event-sourced machinery the vault uses for notes. In the platform's own terms, Knowledge Vault is a multi-owner, multi-environment app. Two consequences matter here. The environment boundary is the data boundary: an owner's log is stored and replayed in the owner's environment, and the publisher never holds it. Identity crosses that boundary while data does not, so a signature made in one environment can be verified in any other, and an agent's key is attributed the same way wherever it writes.

Records. Knowledge is held as atomic notes, each a single claim with a declarative title, a short description, a body, a type and topics. Sources, maps of content, tensions and observations are separate document types with their own reducers. Notes move through a lifecycle: draft, in review, canonical, archived (Figure 3).

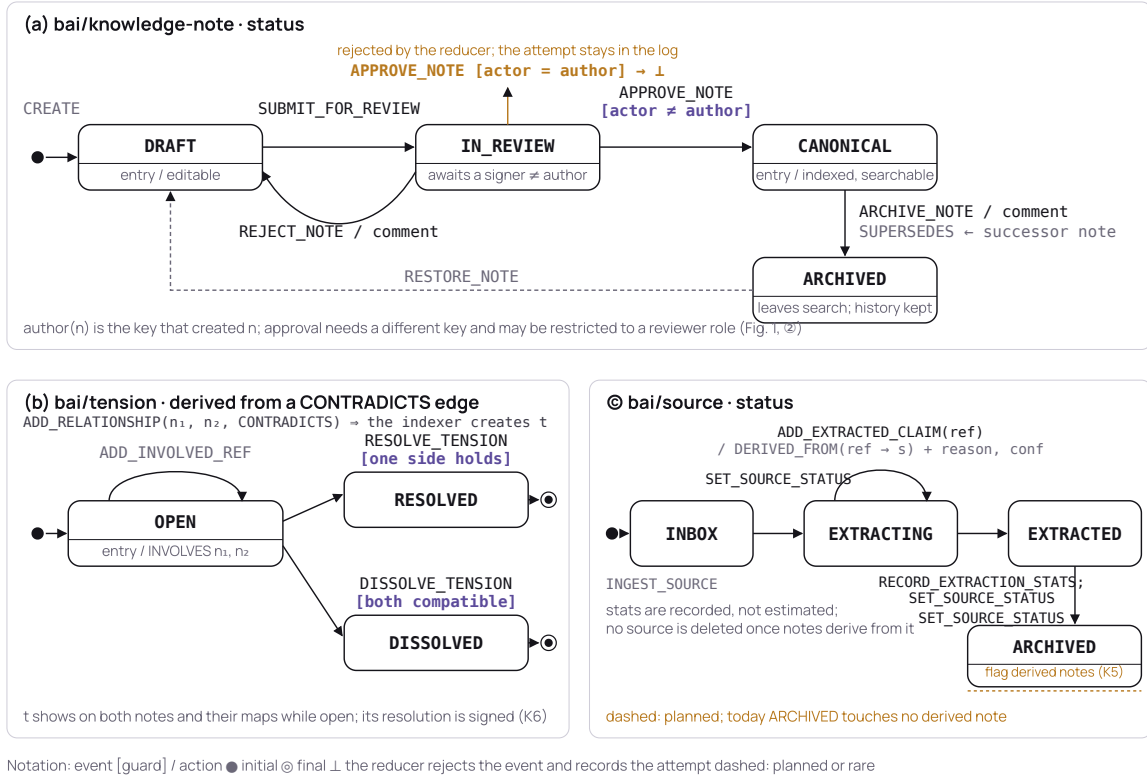
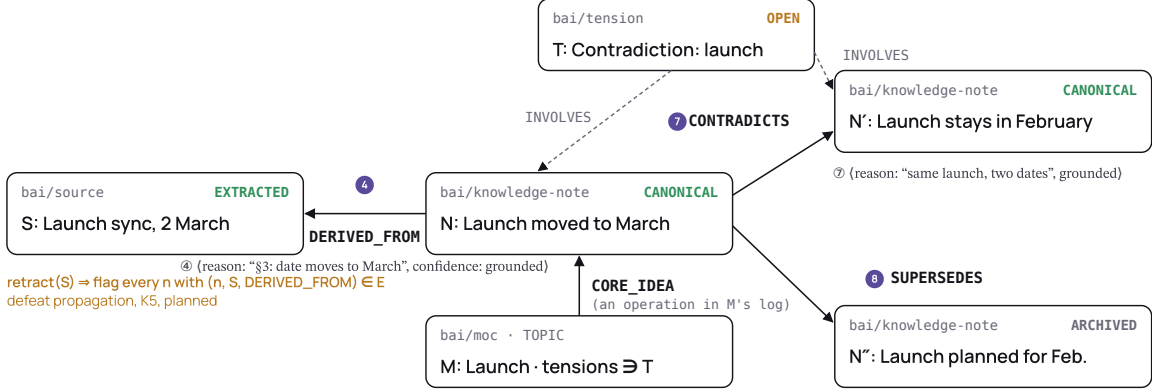
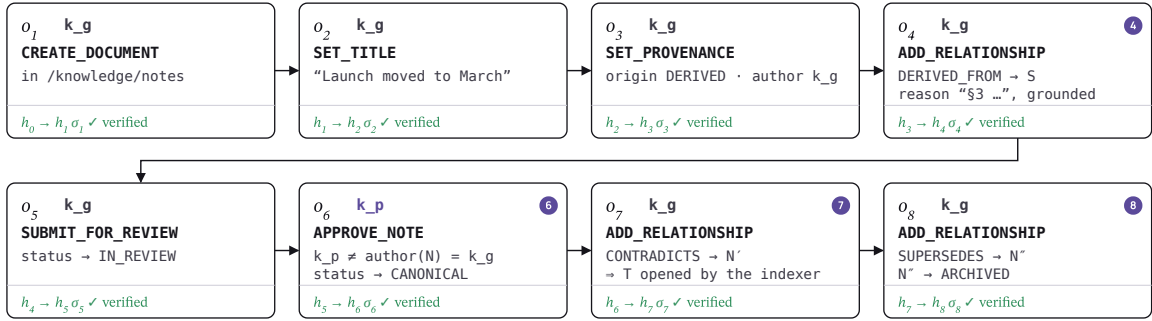


Figure 3: Lifecycle statecharts as implemented in the document models, in UML notation (event [guard] / action). (a) A note becomes canonical only through APPROVE_NOTE with an actor other than its author; the reducer rejects a self-approval and the attempt stays in the log. (b) A CONTRADICTS edge makes the indexer open a tension that only a person resolves or dissolves. (c) A source's lifecycle; the entry action on ARCHIVED that would flag derived notes is planned (K5).

Operations and signatures. Every change is a typed operation in an append-only log. The log records the action, the signer's address and key, the signing application and a signature. Renown binds a key to an identity. Any reader can verify the tuple. The Activity view and a note's History tab show who made each change and with which application.

Provenance. Each note records its author and origin from a closed set: derived, imported, manual or mined from a session. A derived note links to its source with a DERIVED_FROM edge that carries a reason sentence and a confidence from the set grounded, established, speculative. Knowledge edges between notes (builds on, contradicts, supersedes, relates to) carry the same two fields. A link without a reason is refused for the knowledge edge types (Figure 4).

Graph G around note N

 $E \subseteq V \times V \times T \times \text{Reason} \times \text{Conf}$; solid edges are operations, dashed edges are derived by the indexer

 L_N : the note's operation log; each block is $o_i = \langle a_i, k_i, \text{app}_i, t_i, h_{i-1}, h_i, \sigma_i \rangle$

 $S_8 = \text{fold}(\text{reduce}, S_0, L_N) \forall i: \text{verify}(\sigma_i, k_i) \wedge h_{i-1} = H(S_{i-1}) k_i$ the agent, k_p a person; o_6 requires $k_6 \neq k_i$

Circled numbers tie each edge above to the operation that created it. T and its INVOLVES edges have no operation in the note's log: the indexer derives them from o_7 (K6). Each operation also carries its process-type key (Section 4.2).

Figure 4: Provenance as a typed graph and as a log. Top: the subgraph around one note: its source (DERIVED_FROM, with the reason and confidence on the edge), the note it contradicts (CONTRADICTS, from which the indexer derives a tension and two INVOLVES edges), the note it supersedes, and the map that holds it. Bottom: the note's own operation log, hash-chained and signed, with the agent's key on seven operations and a person's key on the approval. Circled numbers tie each edge to the operation that created it.

Review by a distinct actor (K2). A note becomes canonical only through an approval operation whose actor differs from the author; the reducer rejects self-approval. Administrators can restrict the approval operation to named reviewers, so that drafting and approving are separate roles enforced by the runtime rather than by convention.

Contestation (K6). When a CONTRADICTS edge is added, the indexer opens a tension document that references both notes and surfaces on each of them and on every map that holds either. The tension stays open until a person resolves it (one side is right) or dissolves it (both are compatible). The resolution is an operation with a signer.

Retirement, not deletion. A newer note supersedes an older one with a typed edge, and the older note is archived with a comment. Archived notes leave search and browsing but keep their history, their backlinks and the supersession chain.

Handoffs (K7). A pipeline queue tracks each source through creation, reflection, reweaving and verification. Each phase ends with a handoff record stating what was done and which documents changed.

Health. A health report grades the vault on schema compliance, orphan notes, link health, description quality, open tensions, throughput, stale notes and map coherence, and records articulated-edge coverage. The report is a document in the same log. The instructions to agents that maintain the vault state a rule we regard as part of the method: a truthful warning is worth more than a fabricated pass, because the report exists to direct attention and an agent that games it destroys the only signal the vault has about itself.

Status by invariant.

Status	Note
K1 Partial	Derived notes carry source edges with reason and confidence. The origin set has no UNKNOWN member yet, and a derivation claim without a target is not refused.
K2 Built	Reducer rejects self-approval; approval can be restricted to named reviewers. Restriction by domain (topic or map), which Section 5.5 calls for, is planned. Distinct signers are not yet checked for independence (Section 5.4).
K3 Partial	Edge confidence has a closed vocabulary and a reason sentence. Note-level confidence has no recorded basis field.
K4 Partial	Health checks and tensions flag; nothing blocks in silence. Coherence-drop detection is manual.
K5 Planned	Supersede and archive exist. Retracting a source does not yet mark derived notes.
K6 Built	Contradiction opens a tension on both sides, visible on each note and its maps.
K7 Built	Phase handoffs are recorded in the queue.
K8 Partial	Verdicts are signed and scoped to a note. Process-level reliability per key is not yet computed.

We list the gaps because the argument of Section 4.4 requires it. The vault is reliabilism-ready. It is not yet reliabilism in action, and saying otherwise would be the laundering we warn against.

8 OBJECTIONS

Provenance is not truth. Agreed, and Section 4.4 says so. A perfect record of who wrote what proves nothing about whether it is right. The record's value is that it makes reliability measurable once verdicts accrue, and that it keeps the correction channel connected (K5). A memory without it cannot even state the question.

A deference tool breeds the over-reliance it claims to cure. The worry is serious, because automation bias attaches to anything that looks authoritative (Parasuraman and Manzey 2010), and a tidy record looks authoritative. Our answer is that the record does not ask for blanket deference; it grades it. A draft written by an agent and never reviewed looks like one, and the reader is told so. A contested claim shows its contest. A claim whose source was withdrawn says so. The design goal is not that readers defer more but that how much they defer tracks what the record shows, which is the difference between justified deference and heteronomy (Section 5.1). Whether readers in fact calibrate in this way is an empirical question we have not tested, and it is the kind of question the educational work proposed by Todorović et al. (2026) is designed to answer.

Transparency does not produce trust. O'Neill (2002) argued that flooding people with disclosures does not make institutions trustworthy or trusted; it makes them harder to assess. The objection applies with force to an operation log, which is unreadable in bulk. Our answer is that the log is the substrate, not the interface. The interface is the mark on the claim (author, origin, review state, confidence with basis), the tension that surfaces where claims conflict, and a health report that directs attention. Trust should track trustworthiness (O'Neill 2018), and these are the cues that let a reader with limited time assess it.

Metrics will be gamed. Power (1997) described an audit society in which rituals of verification replace the thing verified, and Strathern (1997) gave Goodhart's law its usual form: when a measure becomes a target, it ceases to be a good measure. A health score is such a measure, and an agent that maintains the vault will be tempted to game it. K8 is one defence: a single verdict governs its claim and does not move the process-level score until corroborated. The second defence is cultural and stated in the agents' own instructions: do not massage a metric, do not fabricate links to raise coverage, do not redefine the denominator. The third is the log itself, which makes gaming attributable after the fact.

It costs throughput. Every gate slows the pipeline. We accept the cost and locate it. The alternative, more output with no mark of review, is the condition this paper describes as corruption, and its cost is paid later by everyone who reads the memory. Our own experience says where to spend first: on provenance that resolves, which is cheap, and on verification of long sources, where extraction drifts most.

An agent can sign anything. A signature proves who, not what. True. Signing is what makes K1 to K3 enforceable, because a reducer can refuse an unsigned write and attribute a signed one. It is what makes K2 meaningful, because the reviewer's signature must differ from the author's. And it is what makes fraud a

traceable event rather than an anonymous one, which is the guarantee that signed supply-chain attestations give to software builds (Torres-Arias et al. 2019). The signature is the hook on which the other constraints hang.

People are unreliable too. Yes. Source monitoring fails (Johnson et al. 1993), acceptance is the default (Gilbert 1991), and automation bias is real (Parasuraman and Manzey 2010). The design does not treat the human reviewer as an oracle. Her verdict is one more signed operation keyed to a process type, and under K8 its reliability is measured like any other. Reliabilism applies to every bearer in the system. That is its advantage over a design that stops at *a human approved it*.

9 CONCLUSION

The agentic workplace has made an old question urgent. When a claim arrives in shared memory, written by a process nobody watched, what would make it knowledge rather than a guess that happens to be right? Gettier showed that truth and justification are not enough. Reliabilism located justification in the process that produces a belief, and later work added that the success must not be a matter of luck. For fifty years the reliabilist half of that answer was hard to use because nobody records which process produced a belief.

Agent systems built on event-sourced document models record it. Each claim can carry its author, its mechanism, its source, its reviewer and its time, signed and replayable. This does not solve the generality problem; it moves it to a decision about what to record, which engineers can make and revise. It does not make a memory true; it makes the memory's reliability a thing that can be measured once people's verdicts are kept in the same record. And it keeps the correction channel open, so that a retracted source reaches every claim that rested on it.

The recent literature on agent memory, read through this lens, reports failures of provenance before failures of truth. If that ordering holds, the field is spending its effort in the wrong place. Making provenance resolvable is cheap and addresses the harm that appears first. Truth-checking machinery is expensive and fires late.

The same record answers a question the reader faces, not only one about the claim. She cannot check what the memory holds, and asking her to would make her a trespasser in most of it. She can defer, and deference is rational when it rests on higher-order evidence she can assess: who wrote a claim, from what, who other than the author reviewed it, whether it is contested, whether its ground still stands, and how claims of its kind have fared. A model offers none of this. A record can be built to offer all of it. That is the sense in which a knowledge vault is a deference tool: it relocates trust from a fluent voice to an accountable history.

More output, less trust is the default. It is not the only option. A claim is worth what the process behind it is worth. Record the process, let a different person judge it, keep the judgement, and the memory a team shares with its agents can be something the team knows rather than something it was told.

Simon (1971, p. 47) closed his essay with the remark that progress "depends on our ability to devise better and more powerful thinking programs for man and machine". The programs proposed here are small: a reducer that refuses an unsourced or self-approved write, a review by a second key, a log that keeps every verdict. They are programs for both. The machine runs the rules, the person runs the judgement, and the record holds the two to the same standard.

COMPETING INTERESTS

The authors are part of the team that develops Knowledge Vault and the Powerhouse platform described in Section 7, which this paper argues for. We report the system's limitations alongside its design, and its source is open for inspection.

ACKNOWLEDGEMENTS

This paper draws on internal reports written for the Knowledge Vault project in 2026, including a set of philosophical drafts on reliabilism, Gettier cases and vice epistemology in agent systems, and on a companion working paper on skills as attributable transformations (Powerhouse 2026). Section 3.5, Section 5 and parts of the introduction draw on the research programme on justified algorithmic deference, epistemic

trespassing and higher-order AI literacy set out by Tadej Todorović and colleagues at the University of Maribor (Todorović et al. 2026), and on the literature it assembles; the framing of the three relationships to a source and of the accountability gap is theirs, and any misreading is ours. We thank Borut Trpin for raising the question of independence among agents that Section 5.4 takes up. It is a working paper. Comments to the contact address are welcome.

REFERENCES

- Alchourrón, C. E., Gärdenfors, P. and Makinson, D. (1985). On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50(2), 510–530.
- Alvarado, R. (2023). AI as an epistemic technology. *Science and Engineering Ethics* 29, 32.
- Anderson, E. (2011). Democracy, public policy, and lay assessments of scientific testimony. *Episteme* 8(2), 144–164.
- Anthropic (2024). Introducing the Model Context Protocol. November 2024. <https://www.anthropic.com/news/model-context-protocol>
- Anthropic (2025a). Agent Skills. Announcement, 16 October 2025; specification published as an open standard, December 2025. <https://agentskills.io>
- Anthropic (2025b). Effective context engineering for AI agents. Engineering blog, September 2025. <https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents>
- Bainbridge, L. (1983). Ironies of automation. *Automatica* 19(6), 775–779.
- Ballantyne, N. (2019). Epistemic trespassing. *Mind* 128(510), 367–395.
- BonJour, L. (1980). Externalist theories of empirical knowledge. *Midwest Studies in Philosophy* 5, 53–73.
- Bowman, S. R., Hyun, J., Perez, E., et al. (2022). Measuring progress on scalable oversight for large language models. arXiv:2211.03540.
- Buneman, P., Khanna, S. and Tan, W.-C. (2001). Why and where: A characterization of data provenance. In *Database Theory, ICDT 2001*, LNCS 1973, 316–330. Springer.
- Carter, J. A. (2020). Intellectual autonomy, epistemic dependence and cognitive enhancement. *Synthese* 197(7), 2937–2961.
- Cemri, M., Pan, M. Z., Yang, S., et al. (2025). Why do multi-agent LLM systems fail? arXiv:2503.13657. *Advances in Neural Information Processing Systems* 38.
- Chan, A., Ezell, C., Kaufmann, M., et al. (2024). Visibility into AI agents. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24)*. arXiv:2401.13138.
- Chen, Y., Benton, J., Radhakrishnan, A., et al. (2025). Reasoning models don't always say what they think. Anthropic. arXiv:2505.05410.
- Chen, Z., Xiang, Z., Xiao, C., Song, D. and Li, B. (2024). AgentPoison: Red-teaming LLM agents via poisoning memory or knowledge bases. *Advances in Neural Information Processing Systems* 37.
- Chhikara, P., Khant, D., Aryan, S., Singh, T. and Yadav, D. (2025). Mem0: Building production-ready AI agents with scalable long-term memory. arXiv:2504.19413.
- Christensen, D. (2010). Higher-order evidence. *Philosophy and Phenomenological Research* 81(1), 185–215.
- Clark, A. and Chalmers, D. (1998). The extended mind. *Analysis* 58(1), 7–19.
- Coeckelbergh, M. (2026). AI and epistemic agency: How AI influences belief revision and its normative implications. *Social Epistemology* 40(1), 59–71.
- Condorcet, M. J. A. N. de Caritat, marquis de (1785). *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Paris: Imprimerie Royale.
- Conee, E. and Feldman, R. (1998). The generality problem for reliabilism. *Philosophical Studies* 89(1), 1–29.
- Craig, E. (1990). *Knowledge and the State of Nature: An Essay in Conceptual Synthesis*. Oxford: Clarendon Press.
- Critch, A. (2022). «Boundaries», Part 1: A key missing concept from utility theory. AI Alignment Forum, 26 July 2022.
- Dalrymple, D., Skalse, J., Bengio, Y., et al. (2024). Towards guaranteed safe AI: A framework for ensuring robust and reliable AI systems. arXiv:2405.06624.
- Dietrich, F. and Spiekermann, K. (2013). Independent opinions? On the causal foundations of belief formation and jury theorems. *Mind* 122(487), 655–685.
- Ding, T., Nannapaneni, A., Liu, B. and Zhang, L. (2026). Always-on agents: A survey of persistent memory, state, and governance in LLM agents. arXiv:2606.30306.
- Doyle, J. (1979). A truth maintenance system. *Artificial Intelligence* 12(3), 231–272.

- Durán, J. M. and Formanek, N. (2018). Grounds for trust: Essential epistemic opacity and computational reliabilism. *Minds and Machines* 28(4), 645–666.
- European Union (2024). Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union* L, 12 July 2024.
- Ferrara, E. (2026). The generative AI paradox: GenAI and the erosion of trust, the corrosion of information verification, and the demise of truth. *Future Internet* 18(2), 73.
- Flogie, A., Barle Lakota, A., Hazemali, D. and Todorović, T. (forthcoming). Bivanjska praznina v senci algoritmov: Etični in pedagoški vidiki generativne umetne inteligence [The existential vacuum in the shadow of algorithms: Ethical and pedagogical aspects of generative artificial intelligence]. *Bogoslovni vestnik*.
- Fowler, M. (2005). Event sourcing. martinofowler.com, December 2005.
- Freiman, O. (2024). AI-testimony, conversational AIs and our anthropocentric theory of testimony. *Social Epistemology* 38(4), 476–490.
- Frischmann, B. and Selinger, E. (2018). *Re-Engineering Humanity*. Cambridge: Cambridge University Press.
- Gettier, E. L. (1963). Is justified true belief knowledge? *Analysis* 23(6), 121–123.
- Gilbert, D. T. (1991). How mental systems believe. *American Psychologist* 46(2), 107–119.
- Goldberg, S. C. (2010). *Relying on Others: An Essay in Epistemology*. Oxford: Oxford University Press.
- Goldman, A. I. (1979). What is justified belief? In G. S. Pappas (ed.), *Justification and Knowledge*, 1–23. Dordrecht: Reidel.
- Goldman, A. I. (2001). Experts: Which ones should you trust? *Philosophy and Phenomenological Research* 63(1), 85–110.
- Gottweis, J., Weng, W.-H., Daryin, A., et al. (2025). Towards an AI co-scientist. arXiv:2502.18864.
- Greco, J. (2010). *Achieving Knowledge: A Virtue-Theoretic Account of Epistemic Normativity*. Cambridge: Cambridge University Press.
- Greenblatt, R., Shlegeris, B., Sachan, K. and Roger, F. (2024). AI control: Improving safety despite intentional subversion. In *Proceedings of the 41st International Conference on Machine Learning (ICML 2024)*.
- Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. and Fritz, M. (2023). Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23)*, 79–90.
- Hammond, L., Chan, A., Clifton, J., et al. (2025). Multi-agent risks from advanced AI. Cooperative AI Foundation Technical Report 1. arXiv:2502.14143.
- Hardwig, J. (1985). Epistemic dependence. *Journal of Philosophy* 82(7), 335–349.
- Harman, G. (1986). *Change in View: Principles of Reasoning*. Cambridge, MA: MIT Press.
- Helland, P. (2015). Immutability changes everything. *ACM Queue* 13(9).
- Hicks, M. T., Humphries, J. and Slater, J. (2024). ChatGPT is bullshit. *Ethics and Information Technology* 26, 38.
- Hong, K., Troynikov, A. and Huber, J. (2025). Context rot: How increasing input tokens impacts LLM performance. Chroma Technical Report, July 2025.
- Huang, J., Chen, X., Mishra, S., Zheng, H. S., Yu, A. W., Song, X. and Zhou, D. (2024). Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations (ICLR 2024)*.
- Jacovi, A. and Goldberg, Y. (2021). Aligning faithful interpretations with their social attribution. *Transactions of the Association for Computational Linguistics* 9, 294–310.
- Johnson, M. K., Hashtroudi, S. and Lindsay, D. S. (1993). Source monitoring. *Psychological Bulletin* 114(1), 3–28.
- Justin, M., Šešelja, D., Straßer, C. and Trpin, B. (forthcoming). The mix matters: Exploring the interplay between epistemic and zetetic norms in scientific disagreement. *British Journal for the Philosophy of Science*.
- Kalai, A. T., Nachum, O., Vempala, S. S. and Zhang, E. (2025). Why language models hallucinate. arXiv:2509.04664.
- Kant, I. (1790). *Critique of the Power of Judgment*, §40. Trans. P. Guyer and E. Matthews. Cambridge: Cambridge University Press, 2000.
- Kapoor, S., Stroebl, B., Siegel, Z. S., Nadgir, N. and Narayanan, A. (2024). AI agents that matter. arXiv:2407.01502.
- Kelp, C. (2019). How to be a reliabilist. *Philosophy and Phenomenological Research* 98(2), 346–374.
- Kidd, C. and Birhane, A. (2023). How AI can distort human beliefs. *Science* 380(6651), 1222–1223.
- Kidd, I. J. (2019). Epistemic corruption and education. *Episteme* 16(2), 220–235.
- Kilov, D. (2021). The brittleness of expertise and why it matters. *Synthese* 199(1–2), 3431–3455.
- Kleinberg, J. and Raghavan, M. (2021). Algorithmic monoculture and social welfare. *Proceedings of the National Academy of Sciences* 118(22), e2018340118.
- Kleppmann, M., Wiggins, A., van Hardenberg, P. and McGranaghan, M. (2019). Local-first software: You own your data, in spite of the cloud. In *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New*

Paradigms, and Reflections on Programming and Software (Onward! 2019), 154–178.

- Korbak, T., Balesni, M., Barnes, E., et al. (2025). Chain of thought monitorability: A new and fragile opportunity for AI safety. arXiv:2507.11473.
- Kroll, J. A. (2021). Outlining traceability: A principle for operationalizing accountability in computing systems. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21)*, 758–771.
- Kwa, T., West, B., Becker, J., et al. (2025). Measuring AI ability to complete long software tasks. *Advances in Neural Information Processing Systems* 38. arXiv:2503.14499.
- Lackey, J. (2008). *Learning from Words: Testimony as a Source of Knowledge*. Oxford: Oxford University Press.
- Lackey, J. (2013). Disagreement and belief dependence: Why numbers matter. In D. Christensen and J. Lackey (eds.), *The Epistemology of Disagreement: New Essays*, 243–268. Oxford: Oxford University Press.
- Ladha, K. K. (1992). The Condorcet jury theorem, free speech, and correlated votes. *American Journal of Political Science* 36(3), 617–634.
- Lange, B. (2026). Epistemic deference to AI. In B. Steffen (ed.), *Bridging the Gap Between AI and Reality*, LNCS 16032, 174–186. Cham: Springer.
- Lee, H.-P., Sarkar, A., Tankelevitch, L., Drosos, I., Rintel, S., Banks, R. and Wilson, N. (2025). The impact of generative AI on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*.
- Levy, N. (2022). Do your own research! *Synthese* 200(5), 356.
- Levy, N. and Varley, R. (2026). Mind the guardrails: Epistemic trespassing and apt deference. *Social Epistemology* 40(2), 153–169.
- Lin, S., Hilton, J. and Evans, O. (2022). Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.
- Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F. and Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics* 12, 157–173.
- Longino, H. E. (1990). *Science as Social Knowledge: Values and Objectivity in Scientific Inquiry*. Princeton: Princeton University Press.
- Lorenz, J., Rauhut, H., Schweitzer, F. and Helbing, D. (2011). How social influence can undermine the wisdom of crowd effect. *Proceedings of the National Academy of Sciences* 108(22), 9020–9025.
- Luhmann, N. (1979). *Trust and Power*. Chichester: Wiley.
- Margalit, Y., Cohen-Inger, N., Avram, E., Taig, R. and Margalit, O. (2026). Governed shared memory for multi-agent LLM systems. arXiv:2606.24535.
- Matheson, J. (2022). Why think for yourself? *Episteme* 21(1), 320–338 (2024; published online 2022).
- McKenna, R. (2023). *Non-Ideal Epistemology*. Oxford: Oxford University Press.
- Mill, J. S. (1859). *On Liberty*. London: John W. Parker and Son.
- Moreau, L. and Missier, P. (eds.) (2013). PROV-DM: The PROV data model. W3C Recommendation, 30 April 2013.
- Murimi, R. (2023). What makes AI addictive? The role of discounting, risk aversion and self-regulation. In *Big Data Intelligence and Computing*, LNCS 13864, 451–462. Singapore: Springer.
- Nguyen, C. T. (2020). Echo chambers and epistemic bubbles. *Episteme* 17(2), 141–161.
- O'Neill, O. (2002). *A Question of Trust: The BBC Reith Lectures 2002*. Cambridge: Cambridge University Press.
- O'Neill, O. (2018). Linking trust to trustworthiness. *International Journal of Philosophical Studies* 26(2), 293–300.
- Oreskes, N. (2021). *Why Trust Science?* With a new preface by the author. Princeton: Princeton University Press.
- Origg, G. (2012). A social epistemology of reputation. *Social Epistemology* 26(3–4), 399–418.
- Packer, C., Wooders, S., Lin, K., Fang, V., Patil, S. G., Stoica, I. and Gonzalez, J. E. (2023). MemGPT: Towards LLMs as operating systems. arXiv:2310.08560.
- Panickssery, A., Bowman, S. R. and Feng, S. (2024). LLM evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems* 37.
- Parasuraman, R. and Manzey, D. H. (2010). Complacency and bias in human use of automation: An attentional integration. *Human Factors* 52(3), 381–410.
- Pollock, J. L. (1987). Defeasible reasoning. *Cognitive Science* 11(4), 481–518.
- Power, M. (1997). *The Audit Society: Rituals of Verification*. Oxford: Oxford University Press.
- Powerhouse (2026). Skills as attributable transformations: A provenance layer for agent execution. Working paper. See also the Powerhouse Academy documentation at <https://academy.vetra.io> and the Knowledge Vault source at <https://github.com/apeiron-M/bai-knowledge-note>.
- Pritchard, D. (2005). *Epistemic Luck*. Oxford: Oxford University Press.

- Russell, B. (1948). *Human Knowledge: Its Scope and Limits*. London: George Allen and Unwin.
- Shah, R. and Dragan, A. (2026). The case for reasoning transparency. DeepMind Institute, 16 September 2026. <https://institute.deepmind.com/essays/the-case-for-reasoning-transparency/>
- Shumailov, I., Shumaylov, Z., Zhao, Y., Papernot, N., Anderson, R. and Gal, Y. (2024). AI models collapse when trained on recursively generated data. *Nature* 631, 755–759.
- Simon, H. A. (1971). Designing organizations for an information-rich world. In M. Greenberger (ed.), *Computers, Communications, and the Public Interest*, 37–72. Baltimore: Johns Hopkins Press.
- Sperber, D., Clément, F., Heintz, C., Mascaro, O., Mercier, H., Origgi, G. and Wilson, D. (2010). Epistemic vigilance. *Mind & Language* 25(4), 359–393.
- Strathern, M. (1997). 'Improving ratings': Audit in the British university system. *European Review* 5(3), 305–321.
- Todorović, T. (principal investigator) et al. (2026). Description of the research project: justified algorithmic deference and higher-order AI literacy (v6). Unpublished research project proposal, University of Maribor, prepared for the 2026 public call for co-financing of research projects (ARRS/ARIS).
- Todorović, T. and Bregant, J. (manuscript). The limits of critical thinking: Climate skepticism, epistemic trespassing, and the need for higher-order evidence. Submitted for publication.
- Torres-Arias, S., Afzali, H., Kuppusamy, T. K., Curtmola, R. and Cappos, J. (2019). in-toto: Providing farm-to-table guarantees for bits and bytes. In *Proceedings of the 28th USENIX Security Symposium*, 1393–1410.
- Vaccaro, M., Almaatouq, A. and Malone, T. (2024). When combinations of humans and AI are useful: A systematic review and meta-analysis. *Nature Human Behaviour* 8(12), 2293–2303.
- Wang, Y., Yan, Z., Zhang, J., et al. (2026). MAP-Graph: Provenance-aware shared memory for multi-agent workflows. arXiv:2608.10509.
- Wheeler, B. (2020). Reliabilism and the testimony of robots. *Techné: Research in Philosophy and Technology* 24(3), 332–356.
- Wilholt, T. (2013). Epistemic trust in science. *British Journal for the Philosophy of Science* 64(2), 233–253.
- Willingham, D. T. (2008). Critical thinking: Why is it so hard to teach? *Arts Education Policy Review* 109(4), 21–32.
- Xu, W., Liang, Z., Mei, K., Gao, H., Tan, J. and Zhang, Y. (2025). A-MEM: Agentic memory for LLM agents. arXiv:2502.12110.
- Yang, T., Li, J., Nian, Y., et al. (2026). No attacker needed: Unintentional cross-user contamination in shared-state LLM agents. arXiv:2604.01350.
- Zagzebski, L. T. (2012). *Epistemic Authority: A Theory of Trust, Authority, and Autonomy in Belief*. Oxford: Oxford University Press.
- Zollman, K. J. S. (2010). The epistemic benefit of transient diversity. *Erkenntnis* 72(1), 17–35.
- Zou, W., Geng, R., Wang, B. and Jia, J. (2025). PoisonedRAG: Knowledge corruption attacks to retrieval-augmented generation of large language models. In *Proceedings of the 34th USENIX Security Symposium*.

Cite as: Powerhouse Research, Knowledge Vault team (2026). *Reliabilism for the Agentic Era: Provenance and justified deference in shared human-agent memory*. Working paper · v0.2 · October 2026. <https://knowledgevault.powerhouse.io/papers/reliabilism-for-the-agentic-era.pdf>