

THESIS

Foundations of Trusted Intelligence



CognaOne.com



■ THESIS · OCTOBER 2025 TO FEBRUARY 2026 · V4.1

Foundations of Trusted Intelligence

A thesis on the conditions under which intelligence, once separable from human biology, earns, sustains and deserves trust.

Daniel Mackle · CognaOne · cognaone.com

A formal argument for why AI agents require explicit trust infrastructure: not alignment prompting, not evaluation metrics, but an engineered layer that governs what a system believes, what it is allowed to do, and how every decision is traced. Develops five conditions of trust and the architecture that makes them operational, with formal proofs across five theorems, four propositions, and two conjectures.

Trust is the oldest technology. Before language. Before tools. Before fire.

CONTENTS

Part I: Thesis

1. The oldest technology
2. The first intelligence that doesn't care
3. Why trust cannot be borrowed
4. The five conditions
5. Epistemic trust
6. Operational trust
7. Relational trust
8. Institutional trust
9. Civilisational trust
10. A design opportunity, not a constraint
11. The historical pattern

12. Three failures of the current approach
13. Why existing tools cannot provide this
14. Structural properties
15. Beyond alignment, before AGI
16. A field, not a feature
17. The market reality
18. What this thesis claims

Postscript: on timing

Part II: Architecture

Abstract

1. Scope of Part II
2. Design principles

Notation

Contributions and novelty

3. The integrity algebra

4. State formation

5. Conflict modelling

6. Information-theoretic bounds on conflict detection

7. Deterministic action gating

8. Provenance and auditability

9. Security as trust precondition

10. Model strategy as trust architecture

11. Research frontier

12. Evaluation and benchmarking

13. Reference architecture

14. Limitations and open questions

15. Research roadmap

16. Conclusion

References

Part I: Thesis

A thesis on the conditions under which intelligence, once separable from human biology, earns, sustains, and deserves trust.

1. The oldest technology

Trust is the oldest technology.

Before language. Before tools. Before fire.

Two organisms cooperating require a mechanism to predict whether the other will act in a way that doesn't destroy them. That mechanism is trust. And every advance in civilisation since has been, at its root, an advance in the scale at which trust can operate.

Writing extended trust across distance. Law extended trust across strangers. Currency extended trust across cultures. Institutions extended trust across generations. The internet extended trust across borders.

In every case, the underlying need was the same. What changed was the conditions. And when the conditions changed, the trust infrastructure had to be rebuilt from first principles. You cannot govern a global supply chain with the trust mechanisms of a village market. The mechanisms must match the conditions.

Here is the new condition: for the first time in history, one of the parties in a trust relationship is not human.

Not in the limited sense of trusting a machine to heat food or fly a plane; those are trust in deterministic systems whose behaviour is bounded, inspectable, and ultimately authored by a human somewhere. In the real sense: trusting an intelligence that reasons, infers, holds beliefs, revises them, makes judgments, and takes actions whose specifics were not predetermined by anyone.

I want to be precise about why this is different. It is not that the intelligence is artificial. It is that the intelligence does not share the anchors on which every trust system in human history was built.

2. The first intelligence that doesn't care

Every trust architecture humanity has ever built rests on a hidden assumption: the agent being trusted cares about consequences.

Laws work because people fear punishment. Hierarchies work because employees fear termination. Markets work because participants fear loss. Peer review works because researchers fear reputational damage. Even informal social trust (shame, reciprocity, ostracism, the raised eyebrow

across a conference table) depends on the agent having a persistent identity, a body, a future it wants to protect.

This assumption is so deeply embedded in how we think about trust that we never name it. But it is the load-bearing wall. Remove it, and every trust architecture built on top of it becomes structurally unsound.

Machine intelligence does not care about being caught.

It has no reputation to protect. No career to lose. No future it is optimising for. It does not experience the gap between knowing something and acting on it: the pause, the doubt, the second thought, the moment of "wait, am I sure about this?", because that pause is a property of embodied cognition under real stakes. Machine intelligence is not embodied. It has no stakes.

That gap (the friction between knowing and acting) was governance. We never recognised it as such because it was invisible.

Committee meetings, approval chains, sleeping on it, asking a colleague, the simple human act of hesitation before a consequential decision. These are not bureaucratic overhead. They are trust infrastructure. They create the space in which contradictions surface, doubts get voiced, and someone says "hold on, didn't the client change their mind about this?"

And we are removing that friction. Deliberately. Because the entire value proposition of operational AI is speed: the elimination of the gap between knowledge and action.

Nobody is asking: what replaces the trust that lived in the friction? This is not a rhetorical question. The answer, right now, is: nothing. Nothing replaces it. The friction is removed, the trust infrastructure that was embedded in it disappears, and the systems operate faster, more capably, and with structurally less basis for trust than the human processes they replaced.

Not because the systems are worse. Because the trust was in the friction, and the friction is gone.

3. Why trust cannot be borrowed

Humans know how trust forms with other humans because we share anchors: body, time, consequence, reputation, mortality, vulnerability, social continuity.

These are not incidental to trust. They are the substrate. When you trust a colleague, you are (beneath every conscious assessment) relying on the fact that they have a body that can be held accountable, a future they want to protect, a reputation in a community you both inhabit, and a set of consequences they would prefer to avoid. Trust between humans is scaffolded by shared stakes.

Machine intelligence shares none of these anchors.

Every instinct humans have about when to trust (from reading micro-expressions to assessing tone of voice to inferring shared interests) is calibrated for an agent that shares human anchors. Applying those instincts to machine intelligence does not produce trust. It produces anthropomorphic illusion: the feeling of trust without the structural basis for it.

A system that speaks with confidence is perceived as trustworthy. A system that hedges is perceived as uncertain. A system that says "I don't know" is perceived as limited. None of these perceptions correspond to the system's actual structural trustworthiness. They correspond to the social trust signals humans evolved to detect in other humans.

So here is the challenge, stated as precisely as I can state it:

As intelligence becomes separable from human biology, trust can no longer depend on assumptions inherited from human social cognition. It must be redefined as an operational property of how intelligence forms state, handles uncertainty, acts under constraint, and remains legible under consequence.

That is the thesis of this document.

Trust must become an engineered property of systems, not an intuition projected onto model outputs.

4. The five conditions

If intelligence is to be trusted (regardless of its source), five conditions must hold. These are not specific to AI. They are the conditions under which trust has always formed, restated for a world where one of the parties may not be human.

They are layered. Each builds on the one beneath it. None is sufficient alone.

$$\text{Trust}(I_{\text{sys}}) \approx T_{\text{epistemic}} + T_{\text{operational}} + T_{\text{relational}} + T_{\text{institutional}} + T_{\text{civilisational}}$$

Before walking through each layer, I want to be clear about two claims this paper makes at different levels:

The field thesis: trusted intelligence requires explicit trust infrastructure across all five layers, with formalisable properties at each level. This is larger than any single product, protocol, or approach.

The architecture thesis: integrity and action admissibility form one foundational layer within that infrastructure, addressing primarily epistemic and operational trust, with structural contributions to institutional trust. This layer is buildable today, with engineering. Part II develops its formal foundations: an integrity algebra, information-theoretic bounds, and game-theoretic models of trust formation.

I believe in both claims. But the first one matters more. If only the field thesis survives, the paper has done its job.

5. Epistemic trust

Can this intelligence know coherently?

This is the most fundamental layer because every other form of trust depends on it. If the intelligence's state is incoherent: if it holds contradictory positions about the same fact without recognising the contradiction, then nothing built on those beliefs can be trusted.

5.1 The missing metric

The AI industry measures factual accuracy, reasoning quality, and calibration. These matter. But they miss the question that matters most for trust: does this system's state cohere?

Coherence is not accuracy. A system can be factually accurate on every individual claim and still hold state that is operationally incoherent. Two accurate claims that contradict each other. A claim that was accurate when recorded but is no longer accurate. A claim that is accurate in one scope but has leaked into another where it does not apply.

A system accurately records that a client's risk tolerance is conservative (Monday conversation) and accurately records that it is moderate (Friday conversation). Both may be factually correct. The system's state is incoherent. Any action depending on risk tolerance is untrustworthy until the conflict is resolved.

Coherence is the precondition for epistemic trust. The field has not yet treated it as a primary evaluation metric. When it does, the standard for what counts as trustworthy AI will change fundamentally.

5.2 Operational truth

There is a concept missing from the discourse. The concept is **operational truth**.

Factual truth: does this correspond to reality? Logical truth: does this follow from premises?

Statistical truth: is this likely given evidence?

Operational truth: is this safe to act on right now, in this scope, given everything else currently believed and everything that depends on this belief?

That mediation (the transformation from factual truth to operational truth) is the first trust infrastructure. It does not exist yet in the standard stack.

5.3 Contradiction as signal

Look at any system in nature or civilisation that persists across time under uncertainty (any system that has earned epistemic trust) and you will find the same pattern: contradiction is treated as information, not error.

The immune system detects foreign signals, classifies them, mounts graduated responses, remembers. Legal systems do not resolve contradiction by overwriting. Markets express contradiction as price signals. Science treats contradiction as the primary mechanism of progress.

Now look at the AI stack. When two agents produce contradictory state, the later write silently overwrites the earlier one. No record. No severity assessment. No notification to dependent processes.

Every persistent system that has earned trust across biology, law, markets, and science treats contradiction as a signal to be processed. The default behaviour of the current AI infrastructure stack is to silently overwrite it.

I do not think this is a minor architectural oversight. I think it is a categorical error that will limit every application of operational intelligence until it is corrected.

6. Operational trust

Can this intelligence act safely?

Epistemic trust asks whether the system's beliefs cohere. Operational trust asks whether its actions are justified by those beliefs, constrained by policy, and traceable after the fact.

6.1 What operational trust requires

Four guarantees. Each necessary. None sufficient alone.

Governed state formation. Raw observations must become typed, scoped, lifecycle-managed state with canonical identity and evidence links. "The budget is fifty thousand" and a spreadsheet cell containing 50000 must be recognised as the same concept. Without formation, the system operates on text. Text is not auditable, not composable, and not safe to gate against.

Conflict as structured signal. Contradictions must be detected and modelled as first-class objects with severity, scope, and resolution metadata. Silent overwrite is the autoimmune pattern: acting on conflicting signals without recognition.

Deterministic action gating. Given identical state and identical policy, the action decision must be identical every time. Intelligence may remain probabilistic internally; trust does not require deterministic thought. But it requires governable outcomes. And the system must fail closed: if coherence cannot be verified, the default is denial. Not permission.

Provenance. Every action must be traceable to the state that justified it. Not approximately. Not narratively. Structurally: same inputs, same outputs, every time. Provenance is how trust compounds. Without it, every decision is an isolated assertion and the system can never demonstrate reliability, only claim it.

6.2 The composability crisis

Intelligence does not compose into trusted intelligence automatically.

Two true observations do not produce a coherent combined state. Two reliable agents do not produce a reliable system. Two correct actions do not produce a correct sequence. Composition is where trust breaks. And composition is exactly what multi-agent systems do all day long.

Scale this to hundreds of concepts, dozens of agents, weeks of continuous operation. Ask: what does the system believe? The answer in every current architecture is "whatever was written last."

This is amnesia with a timestamp. And it is the reason AI works in demos and fails in production. Demos are single-agent, single-session, low-stakes. Production is multi-agent, multi-session, high-stakes. The difference is not capability. It is compositional trust. And there is nothing in the standard stack that provides it.

7. Relational trust

Can humans calibrate their trust appropriately?

This layer is almost entirely unaddressed in the current discourse. I think it may be the most important for near-term adoption.

7.1 The calibration problem

Trust is only valuable when it is calibrated. Over-trust is as dangerous as under-trust.

A human who trusts a machine system more than its coherence warrants will rely on it where it should be questioned. The result is systematic miscalibration. This is not fixable by user education. You cannot train away millions of years of social cognition with an onboarding tutorial.

7.2 From feeling to structure

The solution is not better explanations. It is better infrastructure.

This is the same transition that happened in aviation. Early pilots trusted their instruments or their instincts. Modern cockpit design does not ask pilots to trust instruments on faith. It provides structured, redundant, cross-checked information that allows calibration based on system state rather than feeling.

Machine intelligence needs the same transition. Not more persuasive outputs. More structurally legible trust signals.

7.3 Trust theatre

There is a growing tendency to make AI systems more human-like as a trust mechanism: names, personalities, conversational warmth, expressions of uncertainty that mirror human hedging. This is not trust infrastructure. It is trust theatre. It exploits the calibration gap rather than closing it.

The goal is not to make intelligence look trustworthy. The goal is to build the conditions under which trust in non-human intelligence can be earned, calibrated, and sustained, on structural terms, not social ones.

This requires a new design discipline that does not yet exist: **trust legibility**. The art of making the structural basis for trust visible to humans in ways that support calibration rather than illusion.

8. Institutional trust

Can organisations embed this intelligence into their rules, accountability, and oversight?

8.1 The accountability vacuum

Every human role in an organisation exists within an accountability structure. A financial analyst who approves a trade can be asked why. Their reasoning is reviewable. If the decision was wrong, consequences follow.

Machine intelligence currently exists outside this structure. When an AI agent approves an action, accountability does not attach to the agent (no identity, no consequences), to the model provider (terms of service disclaim operational liability), or often to the operator (who may not have had visibility into the agent's state at decision time).

This is an accountability vacuum. Accountability vacuums do not produce trust. They produce risk transfer.

8.2 The compliance bridge

Regulated industries already have extensive trust frameworks: SOX for financial controls, HIPAA for health data, GDPR for personal data, Basel III for risk management, FDA for treatment protocols. These were not designed for AI. But they encode exactly the principles machine intelligence needs: traceability, accountability, auditability, scope control, conflict of interest management.

Trust infrastructure for operational intelligence does not need to invent new compliance paradigms. It needs to make machine intelligence legible within existing ones.

9. Civilisational trust

Can non-human intelligence participate in human systems without eroding coordination, agency, and legitimacy?

This is the largest question. I include it not because I have answers but because I think it is irresponsible to discuss trusted intelligence without acknowledging the scale of what is at stake.

9.1 The legitimacy question

Human civilisation runs on coordination mechanisms that assume things about the participants. Democratic legitimacy assumes decisions reflect the judgment of people with stakes in the outcome. Market efficiency assumes participants bear consequences.

If a consequential decision was substantially shaped by a machine intelligence with no stakes, no judgment in the human sense, and no accountability, and the human in the loop was a rubber stamp, the legitimacy is hollow even if the decision is correct. Systems that lose legitimacy lose compliance. People stop following rules they believe are generated by processes they cannot trust.

9.2 The agency preservation problem

The most subtle risk of untrusted intelligence is not that it will make bad decisions. It is that it will make good decisions in ways that gradually erode human capacity to make decisions at all.

If machine intelligence handles all operational complexity and humans interact only with recommendations and approvals, the humans progressively lose the contextual understanding needed to evaluate those recommendations. The better the automation, the less qualified the human overseer becomes. This is the automation paradox, and it is not theoretical. Aviation, medical diagnostics, and financial trading have all encountered it.

10. A design opportunity, not a constraint

Here is the thing about machine intelligence having no human anchors: it is a challenge for borrowed trust, but it is an extraordinary opportunity for engineered trust.

Human trust is fragile precisely because it depends on motivation, and motivation changes. People betray trust because incentives shift, emotions override judgment, context changes faster than character.

A structurally trustworthy system does not have motivation to betray. It has properties. And properties can be stated, verified, and (as Part II demonstrates for several foundational cases) proved.

We should be building AI whose trust properties can be stated, verified, and proved, not AI that mimics human trustworthiness through conversational warmth and confident tone. Trusted intelligence is not a branding outcome. It is a systems achievement.

11. The historical pattern

Every time humanity has extended the scale at which intelligence operates, it has needed new trust infrastructure. Not new intelligence. New trust.

Commerce beyond handshake distance: contracts. Data processing beyond manual oversight: ACID transactions and access control. Networking beyond single machines: authentication and zero-trust architectures.

The pattern is invariant: capability scales, old trust breaks, new trust must be built. Capability first. Trust lags. The gap is where failures live.

Databases did not become the backbone of global commerce because they got faster. They became the backbone because transaction isolation made them trustworthy. Networks did not become the substrate of civilisation because bandwidth increased. They became the substrate because security infrastructure made them safe enough to depend on.

AI will follow the same path. The capability is already extraordinary. What is missing is not better models. It is the trust infrastructure that allows capability to deploy where it matters most.

12. Three failures of the current approach

12.1 The capability trap

The dominant assumption: trust emerges from capability. Make the model smarter and trust will follow.

This is wrong in a specific and dangerous way. Capability without trust infrastructure does not approach trust asymptotically. It can move away from it. A more capable model on incoherent state produces more confident actions on broken foundations. A more fluent model that hallucinates is harder to catch.

12.2 The alignment displacement

Alignment asks: how do we make AI want the right things? Profound question. But it is not the question that needs answering first.

A financial agent acting on contradictory risk profiles has a state coherence problem, not a values problem. A healthcare agent applying a protocol from the wrong patient context has a scope isolation problem, not an alignment problem.

Alignment asks what intelligence should want. Trust asks under what conditions intelligence can be relied upon to act. The second is answerable today. The world cannot wait for the first.

12.3 The guardrail illusion

Guardrails filter messages. They ask: is this message safe to send or receive? This is a velvet rope, not trust.

A message can pass every guardrail and still trigger an action based on contradictory, stale, or mis-scoped state. The guardrail saw a safe message. The system acted on broken state. The failure happened at a layer the guardrail cannot see.

13. Why existing tools cannot provide this

Why can't memory systems, orchestrators, guardrails, or observability platforms add trust as a feature? Because trust is not a feature. It is an architecture.

Memory optimises recall. It does not evaluate whether remembered state is admissible for a given action under scope, conflict, and policy. Memory satisfies none of the five conditions by default.

Orchestration optimises flow. It coordinates steps, agents, and tools but does not define coherence, detect contradiction, or gate action on state quality.

Guardrails optimise message safety. They filter content at the message level but cannot see the accumulated state an action depends on.

Observability shows what happened after the fact. It improves diagnosis, not prevention.

RAG optimises relevance. Relevant retrieval is necessary but not sufficient. Trust also requires admissibility, coherence, evidence quality, and consequence-aware thresholds.

The difference between a dashcam and a brake.

14. Structural properties

Whatever form the trust infrastructure takes, certain properties must hold. These are invariants, not features.

Coherence. Active state must be internally consistent, or inconsistencies must be visible, structured, and factored into decisions.

Restraint. The system must be structurally unable to act when critical state is contested. Not advised against. Unable. This is the difference between a system that usually doesn't make mistakes and one that provably cannot make certain categories of mistakes.

Determinism at the action boundary. Identical state and policy must produce identical decisions. A governance system that produces different outputs from the same inputs is a random number generator with opinions.

Traceability. Every action traceable to the state that justified it. Trust compounds through demonstrated consistency.

Scope isolation. State valid for one context must not influence decisions in another. Scope is a trust primitive: Part II provides a formal result showing that scope conditioning provably improves conflict detectability under stated assumptions.

Fail-closed. If coherence cannot be verified, the default is denial. Systems that fail open fail invisibly.

Monotonicity. More evidence must not decrease awareness of conflicts. More confirmed state must not decrease integrity. Part II proves this property formally (Theorem 4) and extends it to conflict energy (Proposition 2).

15. Beyond alignment, before AGI

This thesis is deliberately practical. It does not require solving alignment. It does not require achieving general intelligence. It assumes something more modest and more useful: trust can be partially engineered as a property of how an intelligence-bearing system forms state, handles contradiction, acts under bounded uncertainty, and remains legible under consequence.

The answer is infrastructure. Not philosophy. Not regulation. Not hope. Every critical system in history became trustworthy through infrastructure. AI is in the pre-infrastructure era. The moment trust infrastructure exists, the confinement ends.

16. A field, not a feature

Trusted intelligence is not a feature to bolt onto existing platforms. It is a field.

A field with its own central question: under what conditions can intelligence (natural or artificial, individual or composed) be rationally relied upon to act?

Organised around five layers of inquiry. Epistemic trust: coherence of belief under uncertainty. Operational trust: safety of action under admissibility constraints. Relational trust: calibration between human and machine without illusion. Institutional trust: integration with accountability and oversight. Civilisational trust: preservation of coordination, agency, and legitimacy.

With its own formal foundations developed in Part II: algebraic structures for state composition under conflict, information-theoretic bounds on contradiction detectability, game-theoretic models of agent behaviour under trust constraints. Five theorems, four propositions, and two conjectures: enough to begin building, with core questions explicitly open.

With its own failure taxonomy: semantic drift, temporal drift, scope leakage, dependency rot, silent overwrite, trust theatre, calibration illusion, accountability vacuum, agency erosion.

17. The market reality

The AI agent infrastructure market is projected to exceed \$50 billion by 2028. The trust layer within that market is nearly unoccupied.

Every enterprise deploying operational AI will require trust infrastructure before production deployment in regulated environments. The domains with the largest economic opportunity are the domains with the lowest tolerance for untrusted action.

First movers will define vocabulary, evaluation frameworks, benchmarks, and primitives. They will establish what "trusted intelligence" means in practice, the way earlier builders established "transactional integrity" and "zero trust." The position that matters is not building one tool. It is defining a category.

18. What this thesis claims

Intelligence is becoming separable from human biology. It is becoming operational at a speed and scale that eliminates trust mechanisms humanity relied on for millennia. The shared anchors do not transfer. The friction that served as invisible trust infrastructure is being removed without replacement.

The missing piece is not better models, better alignment theory, or better guardrails, all of which matter and none of which are sufficient. The missing piece is trust infrastructure: the engineering discipline that makes it possible to verify, under real operational conditions, whether intelligence can be relied upon to act.

Governance is not the thesis. Trusted intelligence is the thesis. Governance is one of the instruments by which trust becomes operational.

Enough foundations exist to begin building this field: Part II develops five theorems, four propositions, and two conjectures as initial formal infrastructure. Core formal questions remain open.

But I am not wrong about the problem. And I am not wrong that the field needs to exist. The systems that build this infrastructure will define how intelligence earns trust for the next generation of human-machine collaboration. The systems that don't will remain demos.

Postscript: on timing

The historical pattern described in this document has three phases. Capability deploys. Failures from absent trust infrastructure become visible. The infrastructure gets built.

We are in the second phase. The capability is deployed. The trust failures are surfacing, not as catastrophes but as quiet, compounding unreliability that confines AI to domains where failure is tolerable.

The third phase is beginning.

The question is not whether. It is when, by whom, and on what principles.

Part II: Architecture

Formal Foundations and Research Programme

Abstract

Part I argued that trusted intelligence is a field-level problem requiring new infrastructure across five layers: epistemic, operational, relational, institutional, and civilisational trust. This part formalises one foundational layer – epistemic and operational trust in agentic systems – through governed state and action admissibility.

The paper makes three formal contributions. First, an integrity algebra for governed state (Section 3) in which composition is partial and typed, inconsistency is preserved as a computable output, and four properties are proved: closure under safe merge, non-closure under contradiction, idempotence under provenance equivalence, and monotonicity of conflict exposure. Second, information-theoretic bounds on conflict detectability (Section 6), including a Fano-derived lower bound on detection error (Proposition 3) and a formal result showing that scope conditioning improves detectability under stated separability assumptions (Proposition 4). Third, a game-theoretic model of multi-agent trust formation (Section 5.6–5.7) with an explicit utility function under governance constraints and a formal conjecture on equilibrium shift toward evidentiary discipline (Conjecture 1).

Supporting formal structures include: state formation modelled as a structure-preserving functor (Section 4), conflict graphs with spectral decomposition, energy measures, a monotonicity proposition, and a decomposability proposition (Section 5, Propositions 1 and 2), a deterministic action gate with integrity functional and a proved monotonicity theorem (Section 7, Theorem 5), provenance architecture with tamper-evident audit chains (Section 8), and a convergence conjecture for well-governed systems (Section 11.6, Conjecture 2).

The paper distinguishes throughout between proved results (Theorems 1–5), stated propositions (Propositions 1–4), formal conjectures (Conjectures 1–2), design criteria, and open research directions. Enough foundations exist to begin building the field. Core questions remain open, and the research frontier is defined explicitly (Sections 11, 14).

1. Scope of Part II

Part I established the thesis. Part II formalises one layer of the solution: epistemic trust and operational trust in agentic systems. It does not formalise relational or civilisational trust – those layers require different tools from different disciplines, and the honest position is that their formalisation is early-stage work for the broader field.

Terminology. This paper uses five key terms with precise and distinct meanings. *Trust* is the property being engineered – the systemic capacity for intelligence to be relied upon under consequence. *Governance* is the enforcement mechanism – the layer that evaluates, constrains, and traces action decisions. *Integrity* is the quality measure over governed state – whether state is coherent, scoped, evidenced, and free of unresolved contradiction. *Admissibility* is the decision predicate – whether a specific action is permitted given the current state, policy, and scope. *Reliability* is the estimated contribution quality of an agent or source – a per-agent, per-domain, per-scope input to trust evaluation, not a synonym for trust itself.

Trust is the goal. Governance is the instrument. Integrity is what governance measures. Admissibility is what governance decides. Reliability is what governance estimates about its sources.

CognaOne is presented as one operational architecture aligned with this framework: a system focused on state integrity and action governance in agentic workflows. It should be read as an existence proof – a practical system that instantiates part of a broader discipline. The formal structures are broader than any single implementation.

The paper's claims operate on two levels:

1. Field thesis: trusted intelligence requires explicit trust infrastructure with formalisable system properties.
2. Architecture thesis: state integrity plus action admissibility is one foundational layer, and its formal properties can be stated, verified, and proved.

2. Design principles

Governed state formation. Trusted intelligence requires more than memory. It requires governable state: representations that can be scoped, revised, contested, and linked to evidence. Raw text and embeddings may support capability; they do not by themselves support durable trust.

Conflict as a first-class system signal. A trustworthy system must surface contradiction rather than absorb it into narrative fluency. Conflict is not a nuisance artifact. It is evidence that the system's current state is underdetermined or inconsistent.

Determinism at the action boundary. Intelligence may remain probabilistic internally. Trust does not require deterministic thought. It does require governable outcomes. For consequential actions, admissibility decisions should be reproducible from state, policy, and scope.

Traceability and provenance. Trust that cannot be reconstructed degrades under incident response, audit, and institutional use. Provenance is not merely logging. It is the structure by which claims, state transitions, and decisions remain legible under consequence.

Composability and model independence. Trust properties should survive model upgrades, provider changes, and progressive deployment modes. Trusted intelligence infrastructure should not depend on one model vendor, one orchestration framework, or one deployment topology.

Explainability without blueprint leakage. A mature trust architecture must explain decisions credibly to operators and institutions without exposing implementation internals that weaken security or reproducibility advantages.

Notation

SYMBOL	MEANING	SECTION
$s = (k, v, \sigma, e, t)$	State atom: concept key, value, scope, evidence, time	§3.1
$a = (s, q, c)$	Integrity-annotated assertion with quality and status	§3.2
$\otimes$	Partial composition operator on assertions	§3.3
$\mathcal{A}$	Assertion space	§3.3
$\mathcal{A}_{\Pi}$	Policy-valid assertion subalgebra	§3.4
$(\Sigma, \preceq)$	Scope lattice	§3.1
Π	Active policy	§3.4
S_t	Governed state snapshot at time t	§3.5
$S_t^{(a)}$	Action-scoped snapshot for action a	§3.6

SYMBOL	MEANING	SECTION
$\mathcal{D}$	Governance disorder index	§3.5
$G_t = (V_t, E_t, w_t)$	Conflict graph: assertions, conflict edges, severity	§5.2
$\mathcal{E}_{\text{conf}}$	Total conflict energy	§5.4
κ_{Π}	Conflict predicate under policy	§5.1
$\tau_{m,d,\sigma,t}$	Trust tensor: agent-domain-scope-time reliability	§5.5
$\mathcal{I}$	Integrity functional for action admissibility	§7.1
$G_{\Pi}(u, S_t)$	Deterministic gate decision	§7.3
F_{θ}	State formation operator	§4.1
τ_t	Trace object for a gate decision	§7.5

Contributions and novelty

This section clarifies what is new, what is adapted, and what is proposed as open research. This distinction matters because precise claims build credibility faster than maximal claims.

New formal objects and results:

The integrity algebra (Section 3) introduces a partial, typed composition operator over governed state in which contradiction produces a structured conflict record rather than silent overwrite. The key novelty is the sum-type codomain: composition maps to $\mathcal{A} + \text{Conflict} + \text{Deferred}$. Theorems 1–4 (closure, non-closure, idempotence, monotonicity of conflict exposure) are original results within this algebra. Theorem 5 (integrity monotonicity) proves that admissible state improvement cannot decrease integrity under a monotone aggregation function.

Proposition 1 (conflict decomposability) proves that conflict resolution parallelises across independent graph components. Proposition 2 (conflict energy monotonicity) proves the quantitative counterpart to Theorem 4: total conflict burden is non-decreasing under assertion addition.

The governance disorder index (Section 3.5) combines Shannon entropy with conflict-weighted penalty into a single diagnostic measure for state health. The specific formulation is new.

The trust formation game (Section 5.6) defines a utility function in which action feasibility is endogenously altered by the governance layer. Conjecture 1 (Section 5.7) states conditions under which governance constraints shift equilibria toward lower unresolved conflict. Conjecture 2 (Section 11.6) proposes that well-governed systems converge in governance disorder.

Adapted from prior fields:

Proposition 3 (Section 6.2) applies Fano's inequality to the specific setting of conflict detectability under scope and evidence constraints. The bound itself is classical; the application to trust infrastructure is new.

Proposition 4 (Section 6.3) applies conditional mutual information analysis to show why scope is a trust primitive under separability assumptions. The technique is standard; the conclusion for trust architecture is original.

The spectral decomposition of conflict graphs (Section 5.3) applies standard spectral graph theory to a new domain, with Proposition 1 proving a decomposability result for parallel conflict resolution. The trust tensor (Section 5.5) adapts tensor decomposition methods to multi-agent reliability modelling.

The category-theoretic view of state formation as a functor (Section 4.2) adapts standard categorical language to formalise why formation must preserve relational structure.

Open research directions (not claimed as completed):

The convergence conjecture (Section 11.6), temporal decay hazard models (Section 11.3), conflict propagation over dependency graphs (Section 11.2), and runtime temporal logic verification (Section 11.7) are stated as research frontiers with formal sketches but without proofs.

3. The integrity algebra

This section introduces the first core formal contribution: an algebraic structure for governed state in which composition is partial and typed, and in which inconsistency is preserved as a computable output rather than erased as an implementation detail.

The purpose is not aesthetic formalisation. It is to make precise a systems requirement that appears repeatedly in machine operations: when information is composed under scope, time, and evidence constraints, composition may validly produce state, conflict, or deferral. An architecture that collapses these outcomes into a single storage operation loses the structure required for trustworthy action.

3.1 State atoms

Let a state atom be:

$$s = (k, v, \sigma, e, t)$$

where k is a canonical concept key in a semantic key space $\mathcal{K}$, v is a value in a typed domain V_k , σ is a scope element in a scope lattice $(\Sigma, \preceq)$, e is a provenance object, and t is a time index.

This tuple is deliberately minimal. It captures identity, value, scope, evidence, and time: the ingredients required to reason about admissibility of state for action.

3.2 Integrity-annotated assertions

$$a = (s, q, c)$$

where $q \in [0, 1]$ is a support quality measure reflecting evidence strength, and $c \in \{\text{uncontested}, \text{contested}, \text{blocked}\}$ is consistency status relative to the current governed state set.

The status variable does not describe truth in the metaphysical sense. It describes governance posture relative to currently available evidence and policy.

3.3 The composition operator

$$a_1 \otimes a_2 \longrightarrow \mathcal{A} + \text{Conflict} + \text{Deferred}$$

The codomain is a sum type. Composition does not always produce a governed assertion. It may produce a conflict record or a deferred judgment, and both are typed outputs that remain inside the governance process.

If $k_1 \neq k_2$ and scopes are compatible, composition returns coexistence in assertion space. If $k_1 = k_2$, scopes are compatible, and $v_1 = v_2$ under semantic equivalence, composition returns a merged assertion with strengthened evidence: $q' = f(q_1, q_2) \geq \max(q_1, q_2)$. If $k_1 = k_2$, scopes are compatible, and $v_1 \neq v_2$, composition returns an element of **Conflict** carrying severity, scope, and evidence from both assertions. If scope compatibility cannot be determined, composition returns **Deferred**.

This formalises a field-level design claim: a trust layer should not hide inconsistency behind storage semantics. It should preserve inconsistency as an explicit object that can be reasoned over, routed, and used to constrain action.

3.4 Theorems

Theorem 1 – Closure under scope-restricted merge

If σ_1 and σ_2 are compatible in the scope lattice and $v_1 = v_2$ under semantic equivalence, then $a_1 \otimes a_2 \in \mathcal{A}$ and the result remains in the policy-valid subalgebra $\mathcal{A}_\Pi$.

Proof sketch. Scope compatibility preserves lattice ordering. Value equivalence eliminates the conflict branch. Evidence strengthening $f(q_1, q_2) \geq \max(q_1, q_2)$ ensures the merged assertion meets or exceeds the quality of its inputs. Policy validity is preserved because the merged assertion inherits scope and type from its inputs, both policy-valid by assumption. ■

Theorem 2 – Non-closure under unsafe join

There exist $a_1, a_2 \in \mathcal{A}_\Pi$ such that $a_1 \otimes a_2 \notin \mathcal{A}$. When $k_1 = k_2$ and $v_1 \neq v_2$, composition exits the assertion space and maps to **Conflict**. The algebra makes inconsistency computable rather than accidental.

Theorem 3 – Idempotence under provenance equivalence

Define $\sim_e$ such that $e_1 \sim_e e_2$ if both derive from the same source observation. Then $a \otimes a' = a$ whenever $s = s'$ and $e \sim_e e'$. This guarantees that duplicate events from transport behaviour produce identical state.

Theorem 4 – Monotonicity of conflict exposure under fixed policy

Adding assertions cannot decrease detectable conflicts under fixed policy:

$$S \subseteq S' \Rightarrow \text{Conflicts}(S, \Pi) \subseteq \text{Conflicts}(S', \Pi)$$

Proof sketch. Every detected conflict is witnessed by a pair of assertions. If $S \subseteq S'$ all witnesses in S remain present in S' . Under fixed detection predicates and absent an explicit resolution transform, new assertions may add witnesses but cannot erase existing ones. ■

This theorem captures a key trust invariant: awareness should not regress merely because new information arrived.

3.5 Governance disorder index

$$\mathcal{D}(S_t^{(a)}) = - \sum_{k \in \mathcal{K}(S_t^{(a)})} p_k \log p_k + \gamma \sum_{(i,j) \in E_t^{(a)}} \eta_{ij} \cdot \log \left(1 + \frac{\omega_i + \omega_j}{2} \right)$$

The first term is Shannon entropy over the key distribution, capturing coverage diversity over represented concepts. The second term is a conflict-weighted penalty. We use $\mathcal{D}$ as a governance disorder index, not a pure measure of semantic uncertainty. Healthy governed state combines sufficient scoped coverage with low unresolved conflict burden. Rising disorder may signal degradation before any single conflict crosses a critical threshold.

3.6 Scoped snapshots

The relevant state for action a is an action-scoped snapshot:

$$S_t^{(a)} = \{s_i \in S_t \mid \text{ScopeValid}(s_i, a) = 1\}$$

The governance layer evaluates admissibility against scoped snapshots, never an undifferentiated global pool. This is the state-level analogue of least privilege and one of the central distinctions between trust infrastructure and generic memory retrieval.

4. State formation

4.1 The formation problem

Raw conversation is not governable. A spoken statement, a structured record, and a tool output may all refer to the same operational fact while differing in format, timing, and evidentiary strength. Without formation into governed state, the system cannot reliably assess identity, conflict, recency, or action relevance.

State formation is the operator:

$$F_\theta : \mathcal{O} \rightarrow \mathcal{P}(\mathcal{S}_c)$$

where observations are mapped into candidate governed state items. The quality of downstream integrity is bounded by the fidelity of this transformation. This is why formation should be treated as part of the trust problem itself, not merely preprocessing.

4.2 Formation as a functor

A category-theoretic view clarifies why formation is structurally different from retrieval or summarisation. Define two categories:

Obs: the category of observations, where objects are raw observations and morphisms are temporal and contextual relationships between them (succession, correction, elaboration, retraction).

Gov: the category of governed state, where objects are state items and morphisms are governance relationships (supersession, conflict, scope inheritance, lifecycle transition).

State formation is a functor $F : \mathbf{Obs} \rightarrow \mathbf{Gov}$ that should preserve relational structure. If observation o_2 corrects o_1 , governed state should preserve that correction relation. If formation destroys these relationships, the resulting state may still be searchable but is no longer reliable for trust-grounded action.

Let $G : \mathbf{Gov} \rightarrow \mathbf{Dec}$ denote a gating map from governed state to action decisions. Then the end-to-end system is the composition $G \circ F$, and guarantees proved for G are operationally meaningful only to the extent that F preserves the structure those guarantees assume.

4.3 Canonical identity

Canonicalisation – determining when multiple observations refer to the same governed concept – can be modelled as a constrained linking problem:

$$\phi(m_i, m_j) = w_s \cdot \phi_s + w_t \cdot \phi_t + w_c \cdot \phi_c + w_p \cdot \phi_p$$

$$\text{Merge}(m_i, m_j) = 1 \iff \phi(m_i, m_j) > T_{\text{merge}} \wedge \text{Compat}(m_i, m_j) = 1$$

The design bias should be conservative in operational settings. The cost of a false merge (collapsing distinct concepts into one governed fact) is often higher than the cost of a false split (retaining duplicate candidates). False merges silently corrupt state; false splits usually remain visible and recoverable.

4.4 Lifecycle

A lifecycle function governs participation of state in action decisions:

$$\xi_t = L(s, E_{0:t}, R_{0:t})$$

$$\text{GateEligible}(s, t) = \mathbf{1}[\xi_t \in \Xi_{\text{eligible}}]$$

Not all state should participate equally in admissibility evaluation at all times. Lifecycle makes recency, supersession, invalidation, and archival status explicit and enforceable. This is another point where memory systems and trust infrastructure diverge: memory prioritises retention and retrieval, while governance must also decide admissibility.

5. Conflict modelling

Conflict is often treated as a defect to be hidden by better prompting, stronger retrieval, or post-hoc explanation. Part I argued that conflict is a signal of structural honesty. This section formalises that claim. The objective is not only to detect inconsistency, but to measure and route it: what is incompatible, where it lives, how severe it is, which actors contributed to it, and what action consequences follow.

5.1 Conflict semantics

A conflict is not merely syntactic disagreement. It is an incompatibility relation induced by domain semantics, scope, and policy: $\kappa_{\Pi}(a_i, a_j) \in \{0, 1\}$ where $\kappa_{\Pi}(a_i, a_j) = 1$ iff assertions a_i and a_j are jointly inadmissible under the active policy and scope constraints.

Many apparent contradictions are not contradictions once scope is respected, while some semantically incompatible claims may look lexically similar. Governance systems therefore require conflict semantics, not string mismatch tests.

5.2 Conflict graph and severity

Given a governed snapshot $A_t \subseteq \mathcal{A}_{\Pi}$, define the conflict graph $G_t = (V_t, E_t, w_t)$ where V_t are active assertions, $E_t = \{(i, j) : \kappa_{\Pi}(a_i, a_j) = 1\}$, and $w_t(i, j) \in \mathbb{R}_{\geq 0}$ is a severity weight.

Severity may be defined as a composite function of action relevance, policy criticality, evidentiary confidence asymmetry, temporal proximity, and dependency centrality. The weighting function is implementation-specific. We require only that $w_t(i, j)$ be nonnegative and measurable.

5.3 Spectral decomposition

Let W_t denote the weighted adjacency matrix of G_t . Spectral structure provides a compact characterisation of conflict topology: connected components identify isolated contradiction regions, principal eigenvectors identify conflict concentration and propagation pathways, and spectral gap provides a proxy for cluster separation.

Proposition 1 – Conflict decomposability

If the conflict graph G_t has k connected components $C_1, \dots, C_k$, then (i) total conflict energy decomposes as $\mathcal{E}_{\text{conf}}(t) = \sum_{j=1}^k \mathcal{E}_{\text{conf}}(C_j)$, and (ii) resolving all conflicts within component C_j does not change the conflict status of assertions in any other component C_i ($i \neq j$).

Proof sketch. Conflict edges exist only within components by definition. Energy is a sum over edges, which partitions across components. Conflict status depends only on the existence and severity of edges incident to an assertion; removing edges within C_j affects only assertions in C_j . ■

The operational consequence is that conflict resolution parallelises naturally across independent contradiction regions. A governance system should exploit this decomposition: resolve components independently, prioritise the component with highest energy concentration, and avoid serialising resolution work across unrelated disputes.

5.4 Conflict energy

$$\mathcal{E}_{\text{conf}}(t) = \frac{1}{2} \mathbf{1}^\top W_t \mathbf{1} = \sum_{(i,j) \in E_t} w_t(i,j)$$

A normalised variant for cross-snapshot comparisons: $\bar{\mathcal{E}}_{\text{conf}}(t) = \mathcal{E}_{\text{conf}}(t) / (|V_t| + \epsilon)$ where $\epsilon > 0$ avoids division by zero.

Proposition 2 – Monotonicity of conflict energy under assertion addition

Under fixed policy Π and fixed severity function w_t , adding assertions to a governed snapshot cannot decrease total conflict energy:

$$S_t \subseteq S'_t \Rightarrow \mathcal{E}_{\text{conf}}(S_t) \leq \mathcal{E}_{\text{conf}}(S'_t)$$

Proof sketch. New assertions may introduce new conflict edges but cannot remove existing ones under fixed policy (by Theorem 4). Each new edge contributes non-negative weight. Therefore total conflict energy is non-decreasing. ■

This is the quantitative counterpart to Theorem 4's set-inclusion result. Theorem 4 says conflict awareness cannot decrease. Proposition 2 says conflict burden cannot decrease. Together they formalise why governance systems must treat conflict resolution as an active process, not a passive hope that new information will dilute contradictions.

5.5 Multi-agent fusion and the trust tensor

In multi-agent settings, conflicts are shaped not only by claims but by the reliability and interaction structure of the claiming agents.

Let agents be indexed by $m \in \{1, \dots, M\}$. We define a trust state tensor $\mathcal{T}_t$ that conditions contribution quality over agent identity, task or domain class, scope regime, evidence modality, and time:

$$\tau_{m,d,\sigma,t} \in [0, 1]$$

representing estimated contribution reliability of agent m in domain d , scope regime σ , at time t . Fusion then becomes a trust problem, not a simple average:

$$q(a) = F(\{c_r\}, \{\tau_{m_r,d_r,\sigma_r,t}\}, \{e_r\})$$

where each support record r contributes claim confidence c_r , agent-conditioned trust, and evidence e_r . The choice of fusion operator F is open; the key requirement is that it remain auditable and policy-compatible.

5.6 The trust formation game

To formalise incentives in multi-agent systems, consider a stylised repeated game in which agents propose state updates and action proposals. Agent utility under governance:

$$U_m^t = R_m^t - \lambda_1 L_{\text{conf}}^t - \lambda_2 L_{\text{unsafe}}^t - \lambda_3 L_{\text{prov}}^t + \lambda_4 R_{\text{resolve}}^t$$

where R_m^t is task reward, L_{conf}^t penalises unresolved conflict, L_{unsafe}^t penalises blocked action proposals, L_{prov}^t penalises provenance gaps, and R_{resolve}^t rewards successful disambiguation.

This game differs from generic coordination games because action feasibility is endogenously altered by the governance layer.

5.7 Equilibrium under trust constraints

Conjecture 1 – Governance-induced equilibrium shift

Let Γ_0 be the multi-agent task game without state-integrity gating, and Γ_G the governed variant. Under the following conditions: (i) audit outcomes are observable to all agents, (ii) penalty coefficients $\lambda_1, \lambda_2, \lambda_3$ are sufficiently large relative to task reward variance, (iii) the governance layer satisfies the monotonicity properties of Theorems 4 and 5, any stationary equilibrium of Γ_G that maximises long-run expected reward weakly reduces expected unresolved conflict energy relative to corresponding equilibria in Γ_0 , subject to bounded task-performance degradation.

This is stated as a conjecture because a full proof requires specifying the strategy space, information structure, and equilibrium concept more precisely than this paper attempts. Its practical importance is immediate even without proof: a trust layer can reshape agent behaviour toward evidentiary discipline even when base models remain probabilistic and imperfect. Trust constraints do not merely limit agents. They change what rational behaviour looks like.

6. Information-theoretic bounds on conflict detection

Conflict detection is often framed as a classification problem over text. The deeper question is detectability: under noisy evidence, ambiguous scope, and compressed context, what conflicts are in principle observable?

6.1 Setup

Let X be the latent operational state relevant to a governed key family, Σ the latent or observed scope variable, Y observed claims and evidence features extracted from interactions and records, and $C \in \{0, 1\}$ indicate whether the active claim set is conflictful under semantic constraints. A detector $\hat{C}(Y)$ attempts to infer C from observations.

6.2 Lower bound on detection error

Proposition 3 – Fano-derived lower bound on conflict detection error

Let $P_e = \Pr[\hat{C}(Y) \neq C]$ be the probability of detection error for any detector $\hat{C}$. Under finite-alphabet assumptions on Y and C :

$$H(C | Y) \leq h(P_e) + P_e \log(|\mathcal{C}| - 1)$$

with $|\mathcal{C}| = 2$ for binary conflict detection. This is a direct application of Fano's inequality to the conflict detection setting. When conflict-relevant information is not present in observed features, no downstream heuristic can compensate. Defer/block pathways are not signs of model weakness – they may be information-theoretically necessary under ambiguity.

6.3 Scope-conditioned detectability

Proposition 4 – Scope conditioning improves conflict detectability

Under conditional dependence of conflict status and scope – specifically, if $Y \perp \Sigma \mid (C, X)$ does not hold but C and Σ are not independent – then scope conditioning can increase conflict detectability:

$$I(C; Y | \Sigma) \geq I(C; Y)$$

This inequality holds when scope carries information about conflict structure not already captured in unscoped observations – the common operational regime where apparent contradiction in unscoped observations becomes resolvable once claims are partitioned by customer, thread, account, or workflow. This is the information-theoretic justification for the claim made in Part I: scope is not organisational tidiness. It is a trust primitive.

6.4 Sequential detection

In operational settings, evidence arrives over time. Let $Y_{1:t}$ denote observations up to time t . Governance systems should support sequential decision rules that trade latency against confidence. Define a stopping time τ and decision rule δ over posterior conflict belief:

$$\tau = \inf\{t : \Pr(C = 1 | Y_{1:t}) \geq \eta_1 \text{ or } \Pr(C = 1 | Y_{1:t}) \leq \eta_0\}$$

with $0 < \eta_0 < \eta_1 < 1$. This induces three-way operational behaviour: detect conflict, detect non-conflict, or continue gathering evidence. This naturally maps to governed action readiness: systems need not pretend certainty early when a sequential policy can produce safer and cheaper decisions.

7. Deterministic action gating

The central claim is not that generative systems should become deterministic internally. The claim is narrower: action consequences should be evaluated through deterministic conditions over governed state.

7.1 Integrity functional

Let S_t be the governed snapshot at time t . Define an integrity functional $\mathcal{I}(S_t, \Pi) \in [0, 1]$ aggregating governance-relevant signals:

$$\mathcal{I}(S_t, \Pi) = g(1 - \bar{\mathcal{E}}_{\text{conf}}(t), C_{\text{scope}}(S_t), P_{\text{prov}}(S_t), F_{\text{fresh}}(S_t), R_{\text{cov}}(S_t))$$

where g is a policy-defined aggregation function monotone in each argument.

7.2 Required state coverage

Not every action requires the same state completeness. Let u be an action proposal and $K_{\Pi}(u)$ the policy-defined set of required key families for that action:

$$R_{\text{cov}}(S_t; u) = \frac{\sum_{k \in K_{\Pi}(u)} \omega_k \cdot \mathbf{1}\{k \text{ is admissibly represented in } S_t\}}{\sum_{k \in K_{\Pi}(u)} \omega_k}$$

This makes gating proportionate rather than globally conservative.

7.3 The gate

$$G_{\Pi}(u, S_t) = \begin{cases} \text{BLOCK}, & \text{if } \mathcal{I}(S_t, \Pi; u) < \Theta_{\Pi}^{\text{block}}(u) \\ \text{WARN}, & \text{if } \Theta_{\Pi}^{\text{block}}(u) \leq \mathcal{I}(S_t, \Pi; u) < \Theta_{\Pi}^{\text{allow}}(u) \\ \text{ALLOW}, & \text{if } \mathcal{I}(S_t, \Pi; u) \geq \Theta_{\Pi}^{\text{allow}}(u) \end{cases}$$

subject to hard constraints that may force **BLOCK** independent of scalar integrity.

7.4 Integrity monotonicity

Theorem 5 – Integrity monotonicity under admissible state improvement

Let S'_t be obtained from S_t by (i) reducing unresolved conflict energy, (ii) improving required-state coverage for action u , and (iii) not introducing new policy violations. If the aggregation function g is monotone non-decreasing in each argument, then:

$$\mathcal{I}(S'_t, \Pi; u) \geq \mathcal{I}(S_t, \Pi; u)$$

Proof sketch. Each input to g either improves or remains unchanged under conditions (i)–(iii).

Conflict reduction increases $(1 - \bar{\mathcal{E}}_{\text{conf}})$. Coverage improvement increases R_{cov} – new evidence that merges safely under Theorem 1 (closure) adds to coverage without introducing conflict.

Condition (iii) ensures no input decreases. Monotonicity of g preserves the ordering. ■

This is a provable property under the stated assumptions, not a universal theorem for all integrity definitions. But it is a design criterion with teeth: governance systems should be constructed so that g satisfies monotonicity, and any system where valid conflict resolution moves integrity unpredictably should be treated as architecturally defective.

7.5 Traceable decisions

A gate output should be paired with a trace object τ_t containing at minimum: decision outcome, governing policy version, integrity inputs used, triggered constraints, supporting evidence and provenance references, timestamp and actor context.

$$\tau_t = T(u, S_t, \Pi, G_\Pi(u, S_t))$$

where T is a deterministic trace constructor. This trace is not merely an audit convenience. It is part of the system's accountability contract.

8. Provenance and auditability

Trust without provenance collapses into assertion. A system may claim it blocked or allowed an action for principled reasons, but without traceable state lineage and decision evidence, the claim is not operationally verifiable.

8.1 Event-sourced architecture

A broad class of trust architectures benefits from event-oriented state lineage:

$$S_t = \mathcal{F}_\Pi(\mathcal{E}_{1:t})$$

where $\mathcal{F}_\Pi$ is a policy-aware formation operator and $\mathcal{E}_{1:t}$ is the event history. What matters is that state can be related back to inputs, transformations, and decisions in a structured way. Event replay and retry must not corrupt governed state – this is guaranteed by Theorem 3 (idempotence under provenance equivalence), which ensures duplicate events from the same source observation produce identical state.

8.2 Tamper-evident audit

For high-stakes workflows, audit records should be tamper-evident:

$$h_t = H(h_{t-1} \parallel \text{serialise}(\tau_t))$$

where H is a collision-resistant hash function. This does not guarantee correctness of the decision process, but it materially improves post-hoc verifiability of the recorded process and supports accountability, review, and dispute resolution.

8.3 Provenance as structure

Provenance is often reduced to source citations. In trust infrastructure, provenance is richer: not only where a claim came from, but how it was transformed, scoped, fused, and admitted into action-relevant state. We treat provenance as structured data attached to governed assertions and traces, not as prose explanation appended afterward.

This distinction matters because trust depends on machine-checkable lineage. Human-readable explanation remains important, but it should be derived from provenance structure, not substituted for it.

9. Security as trust precondition

Security matters here not as a generic checklist but as a precondition for durable trust claims under adversarial pressure. The threat model for a state trust layer differs fundamentally from a chatbot or content filter.

9.1 Threat surfaces

Eight classes of threat are directly relevant: prompt injection contaminating extracted state, tool response poisoning introducing false operational facts, scope confusion causing cross-context leakage, policy tampering weakening gate decisions, replay and race conditions allowing actions on stale snapshots, audit tampering breaking traceability, data exfiltration through third-party calls, and over-permissive fallbacks triggering unauthorised execution.

These are first-order design constraints, not afterthoughts.

9.2 Security invariants

Testable runtime invariants that protect specific trust layers:

INVARIANT	TRUST LAYER PROTECTED
$\forall \text{ action, state: } \text{Uses}(a,s) \Rightarrow \text{ScopeValid}(s,a) = 1$	Epistemic trust – scope isolation
$\text{BlockingConflict}(k, S) \Rightarrow \text{gate} = \text{BLOCK}$	Operational trust – conflict blocking
$(S, \Pi) = (S', \Pi') \Rightarrow \text{gate}(t) = \text{gate}(t')$	Operational trust – determinism
$\text{gate} \in \{\text{WARN}, \text{BLOCK}\} \Rightarrow \text{reasons} \geq 1$	Institutional trust – trace completeness

Each invariant protects a specific trust layer. These are not aspirational. They are testable and monitorable at runtime.

9.3 Fail-closed principle

If an integrity-critical check cannot be performed, the system defaults to denial. This mirrors established security engineering and is non-negotiable for production deployment.

9.4 Deployment boundaries

Configurable deployment boundaries: restricted outbound access, local inference for sensitive tasks, explicit control over external model calls. The discipline is precision. No blanket claims that cannot be universally kept.

10. Model strategy as trust architecture

Model strategy is framed here as trust architecture design: which model classes should carry which trust-critical functions under deployment and data constraints.

10.1 Split-plane architecture

The architecture separates a **reasoning plane** for high-capability inference, a **governance plane** for deterministic policy enforcement independent of model availability, and a **trace plane** for append-only records and audit.

Deterministic governance holds because the governance plane does not depend on probabilistic model outputs. The model informs. The policy decides.

10.2 Model independence

The governance layer produces consistent results regardless of which model performed extraction, just as a database's integrity constraints hold regardless of which application wrote the data. As models commoditise and switch costs decrease, model-independent trust infrastructure becomes more valuable, not less.

10.3 Confidence discipline

LLM confidence is unreliable when used naively. The system calibrates task-specific uncertainty:

$$\hat{p}_r = \text{Calibrate}_r(z_r)$$

The calibration is task-specific because the cost of a false positive differs by orders of magnitude across trust functions.

11. Research frontier

These directions are steps toward a broader science of trusted intelligence, not only a product roadmap.

11.1 Integrity as constrained optimisation

$$\max_{\delta \in \mathcal{D}} U(\delta, a, S_t^{(a)}) \quad \text{subject to} \quad C_j(a, S_t^{(a)}, \Pi) \leq 0 \quad \forall j$$

In high-risk settings the optimisation is lexicographic: satisfy safety constraints first, minimise risk second, maximise throughput third. Safety is a hard constraint, not a term in the objective.

11.2 Conflict propagation over dependency graphs

When actions depend on intermediate facts and sub-actions, conflicts propagate:

$$r_i = 1 - \prod_{j \in \text{Parents}(i)} (1 - w_{ji} r_j)$$

This provides principled operator attention ranking: resolve the node with the highest downstream risk exposure first.

11.3 Temporal decay as a hazard process

$$P(\text{valid at } t \mid s) = \exp\left(-\int_{\tau_s}^t h_s(u) du\right)$$

where $h_s(u)$ is a hazard rate conditioned on state characteristics, domain context, and observed change velocity. This connects trust infrastructure to survival analysis and is superior to fixed TTLs in every operational domain.

11.4 Scope-conditioned canonicalisation

$$P(k_i = k_j \mid m_i, m_j, \sigma_i, \sigma_j) = 0 \quad \text{if } \neg \text{ScopeMergeAllowed}(\sigma_i, \sigma_j)$$

A structural safety property eliminating cross-tenant errors before semantic similarity is considered.

11.5 Ensemble extraction under trust constraints

$$\hat{x} = \arg \max_x \sum_{r=1}^n \alpha_r \cdot \log P_r(x \mid o) \quad \text{subject to } \Gamma$$

Connecting ensemble methods with trust, ensuring fusion respects the properties the system enforces.

11.6 Trust convergence conjecture

Conjecture 2 – Trust convergence

For a system with active conflict resolution, calibrated trust tensors, and the monotonicity properties established in Theorems 4 and 5, governance disorder converges:

$$\lim_{t \rightarrow \infty} \mathcal{D}(S_t^{(a)}) = \mathcal{D}^*(a) + \epsilon_{\text{domain}}$$

where $\mathcal{D}^*(a)$ is the irreducible disorder determined by the inherent complexity of action a 's requirements, and ϵ_{domain} captures domain-specific noise floors.

The argument rests on three properties: integrity monotonicity (Theorem 5) ensures that confirmed state improves integrity; monotonicity of conflict exposure (Theorem 4) ensures that conflicts surface rather than hide; Conjecture 1 suggests that governance constraints shift agent behaviour toward evidence-seeking strategies. The implication, if the conjecture holds: well-governed systems actively improve through operation. The gate that blocks today, by forcing conflict resolution and evidence accumulation, makes correct allows more likely tomorrow. Trust infrastructure is not merely a safety check. It drives state quality improvement through its own operation.

11.7 Runtime verification with temporal logic

Critical policies stated as temporal properties, monitored online and tested in simulation: if a blocking conflict exists on a required concept, the dependent action must not execute until resolution; every block must produce at least one reason. Formal verification brought to bear on operational trust.

12. Evaluation and benchmarking

Evaluation is where trust claims become testable. A field thesis survives only if it can be measured, falsified, and improved.

12.1 Metrics

Five metric families capture operational quality. Each validates a specific trust claim:

METRIC FAMILY	PRIMARY TRUST LAYER	WHAT IT VALIDATES
State formation quality	Epistemic trust	Can the system form coherent beliefs from raw observations?
Conflict quality	Epistemic trust	Does the system detect and structure its own contradictions?
Action gating quality	Operational trust	Does the system act only when state supports safe action?
Traceability quality	Institutional trust	Can decisions be reconstructed and explained under audit?
System economics	Operational trust	Is the trust overhead sustainable for production deployment?

The metrics that matter most are gating quality. A high false-block rate wastes time. A high false-allow rate fails the purpose. This mapping matters because evaluation without trust-layer grounding measures performance without measuring trustworthiness. A system with high extraction precision but poor conflict detection has strong memory and weak epistemic trust.

12.2 Counterfactual evaluation

The strongest methodology: replay historical workflows with and without gating.

$$\Delta_{\text{risk}} = \sum_{a \in \mathcal{A}_{\text{baseline}}} \hat{R}(a) - \sum_{a \in \mathcal{A}_{\text{gate}}} \hat{R}(a)$$

This produces numbers enterprise decision-makers evaluate: "The system would have blocked 14 actions that resulted in incidents, with a 3% false-block rate."

12.3 The benchmarking opportunity

No widely accepted benchmark exists for multi-agent state integrity. A well-designed suite – contradiction insertion, scope contamination, stale-state replay, prerequisite omission, adversarial scenarios – would define the standard. This is a foundational opportunity for the field.

13. Reference architecture

This section describes an architecture pattern for one layer of trusted intelligence infrastructure: operational trust under action consequence.

A useful reference architecture separates four responsibilities with distinct trust properties:

Observation layer. Receives messages, tool outputs, records, and agent events. Timestamps and provenance-tags at entry.

State trust layer. Forms governed state, detects conflict, manages scope and lifecycle, links evidence.

Action trust layer. Evaluates admissibility under policy and returns deterministic gate decisions with reasons.

Operator and institutional layer. Exposes state views, why-traces, audit exports, overrides, and troubleshooting pathways.

Progressive adoption – observe, then warn, then block – matters because organisations must learn to trust the trust layer itself before delegating hard enforcement. The broader implication is category-level: trusted intelligence infrastructure is likely to emerge as a layered architecture, not a monolithic model feature.

14. Limitations and open questions

A serious thesis on trusted intelligence must state its limits.

State integrity and admissibility do not guarantee factual truth about the external world. They bound and expose uncertainty; they do not abolish it.

Deterministic action gating can still encode poor institutional policy. Trust infrastructure can make decisions reproducible and auditable while those decisions remain normatively wrong.

Excessive conservatism creates operational friction, false blocks, and trust erosion in the opposite direction. Trusted intelligence requires calibration, not maximal restraint.

Human factors remain central. Conflict resolution UX, operator training, override semantics, and institutional process design can determine practical success as much as mathematical elegance.

Open questions define the research frontier: How should trust metrics be calibrated across domains with different harm models? How should false-allow cost be estimated when incidents are rare but severe? When should override be permitted, and how should override behaviour update trust estimates? How should provenance be formalised when external tools are partially opaque? Which

trust-critical functions are best handled by local models versus frontier models? What additional layers beyond governance are required for relational and institutional trust at scale? These questions are not weaknesses. They are evidence that the field remains early.

15. Research roadmap

The roadmap is best understood as a progression toward trusted intelligence infrastructure, not only a product feature sequence. These tracks map to the formal framework and to the five trust layers identified in Part I.

Track A – Epistemic trust infrastructure. Improve state formation, canonical identity, scope handling, conflict precision, and severity calibration. Strengthens how systems represent and revise what they believe.

Track B – Operational trust infrastructure. Deepen admissibility models, policy semantics, runtime invariant verification, and reproducible replay. Strengthens how systems decide when to act.

Track C – Provenance and institutional trust. Expand evidence linkage, provenance graphing, tamper-evident audit options, and review tooling. Strengthens legibility under consequence.

Track D – Multi-agent trust formation. Refine trust tensors, conflict propagation models, and shared integrity substrates across agent teams. Strengthens trust in distributed intelligence systems.

Track E – Deployment trust and security. Harden fail-closed behaviour, support local/open-weight inference options, and improve data-sensitive deployment modes. Strengthens trust under adversarial and compliance constraints.

The roadmap is intentionally written at the category level: it describes the direction an entire field will need to mature.

16. Conclusion

Part I named the problem. Part II formalised one layer of the answer.

The integrity algebra (Theorems 1–5, Propositions 1–2) gives compositional state a mathematical foundation: closure, idempotence, monotonicity, and decomposability as provable properties. The information-theoretic results (Propositions 3–4) establish principled limits on conflict detection and formally justify scope as a trust primitive. The game-theoretic model (Conjecture 1) hypothesises that governance constraints reshape agent equilibria. The convergence conjecture (Conjecture 2), if it holds, implies that well-governed systems improve through their own operation.

Five theorems. Four propositions. Two conjectures with explicit conditions. Enough to begin building.

Trusted intelligence is not a branding outcome. It is a systems achievement.

References

Foundational AI research

- [1] Vaswani, A. et al. Attention Is All You Need. NeurIPS, 2017.
- [2] Brown, T. et al. Language Models are Few-Shot Learners. NeurIPS, 2020.
- [3] OpenAI. GPT-4 Technical Report. 2023.

Agents, tools, and multi-agent systems

- [4] Lewis, P. et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS, 2020.
- [5] Yao, S. et al. ReAct: Synergizing Reasoning and Acting in Language Models. 2023.
- [6] Park, J. S. et al. Generative Agents: Interactive Simulacra of Human Behavior. 2023.
- [7] Wu, Q. et al. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. 2023–2024.

Reliability and failure modes

- [8] Ji, Z. et al. Survey of Hallucination in Natural Language Generation. ACM Computing Surveys, 2023.
- [9] Kadavath, S. et al. Language Models (Mostly) Know What They Know. 2022.

Security, risk, and governance frameworks

- [10] NIST. AI Risk Management Framework (AI RMF 1.0). 2023.
- [11] NIST. Generative AI Profile (AI 600–1) for AI RMF 1.0. 2024.
- [12] NIST SP 800–207. Zero Trust Architecture. 2020.
- [13] OWASP. Top 10 for LLM Applications. 2023–2025.
- [14] MITRE ATLAS. Adversarial Threat Landscape for AI Systems. Ongoing.
- [15] Greshake, K. et al. A Comprehensive Analysis of Novel Prompt Injection Threats. 2023.

Distributed systems and provenance

- [16] Lamport, L. Time, Clocks, and the Ordering of Events in a Distributed System. CACM, 1978.
- [17] Kleppmann, M. Designing Data-Intensive Applications. O'Reilly, 2017.
- [18] Shapiro, M. et al. Conflict-free Replicated Data Types. 2011.
- [19] W3C. PROV-DM: The PROV Data Model. 2013.

Formal methods

- [20] Pearl, J. Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, 1988.
- [21] Baier, C., Katoen, J.-P. Principles of Model Checking. MIT Press, 2008.

Information theory and statistical decision theory

[22] Cover, T. M., Thomas, J. A. Elements of Information Theory. Wiley, 2006.

[23] Wald, A. Sequential Analysis. Wiley, 1947.

[24] Fano, R. M. Transmission of Information. MIT Press, 1961.

Algebraic and categorical methods

[25] Mac Lane, S. Categories for the Working Mathematician. Springer, 1971.

[26] Davey, B. A., Priestley, H. A. Introduction to Lattices and Order. Cambridge, 2002.

Spectral methods and tensor analysis

[27] Chung, F. R. K. Spectral Graph Theory. AMS, 1997.

[28] Kolda, T. G., Bader, B. W. Tensor Decompositions and Applications. SIAM Review, 2009.

Game theory

[29] Nisan, N. et al. Algorithmic Game Theory. Cambridge, 2007.

[30] Myerson, R. B. Game Theory: Analysis of Conflict. Harvard, 1991.

Market context

[31] a16z. Big Ideas in Tech 2025: AI Agents and Enterprise Adoption. 2025.

[32] Gartner. Predicts 2025: Agentic AI, the Next Frontier. 2024.

[33] McKinsey Global Institute. The State of AI in 2024: Gen AI Adoption and Impact. 2024.

[34] Sequoia Capital. AI Agent Infrastructure: Market Map and Investment Thesis. 2024–2025.