The WisdomTwin Trust Layer

A Testable Control Specification for Role-Specific AI Decision Support

Roman Bodnarchuk | WisdomTwin, Inc. | ORCID 0009-0004-3113-2118

WT-200-001 | Version 1.0 | September 24, 2026

Status of record. Working paper; not peer reviewed. This is a proposed control specification, not a description of verified product functionality or regulatory certification. No enterprise deployment, security evaluation, or human-participant study was conducted for this paper. No DOI has been assigned.

Abstract

A role-specific AI assistant creates value only if an authorized recipient can use its evidence without mistaking a generated recommendation for the authority of an officeholder. This paper translates the judgment-latency framework of WT-100-001 into a proposed Trust Layer: a set of enforceable boundaries around data admission, role memory, retrieval, response delivery, and delegated action. The design treats authorization as specific to the requesting person, intended recipient, source, purpose, time, and requested action. It separates historical precedent from current policy and decision authority, requires safe decline when evidence or permissions are uncertain, and specifies an audit record without requiring disclosure of private model reasoning. Voice meetings and multi-party conversations receive explicit audience controls. A traceability matrix connects principal failure modes to testable controls and release evidence. The contribution is a falsifiable specification for future implementation and evaluation. No claim is made that the proposed controls have been implemented, independently validated, or shown to satisfy any particular legal regime.

Keywords: role-specific AI; authorization; provenance; decision rights; enterprise memory; human oversight; auditability; judgment latency.

1. Problem and contribution

WT-100-001 defines judgment latency as the interval between a decision being ready and substantive attention by the authorized role or forum [1]. It proposes that bounded AI decision support could shorten that interval while also introducing error, disclosure, and review burden. The present paper makes the governance part of that proposition inspectable. It specifies what must be true before an assistant may answer, deliver an answer to an audience, or initiate a delegated action.

The central distinction is between possessing information and being entitled to use it in a particular interaction. A system may have indexed a former CFO's notes without having permission to reveal them to a current employee. A current CFO may be able to read a confidential acquisition file without having authority to disclose it in a routine management meeting. An assistant that reproduces the apparent style of either CFO has not thereby acquired their professional authority.

This paper contributes a proposed control model, an authority vocabulary, an audit schema, and acceptance evidence for future testing. These are author-proposed design requirements. They are not findings about deployed WisdomTwin systems, universal legal requirements, or claims that the design reconstructs an individual's entire judgment.

2. Method and boundaries

The method is requirements synthesis and threat-informed design. It starts from WT-100-001, uses the NIST AI risk-management resources as governance context [2,3], and checks a concrete permission-integration example against Microsoft documentation [4]. It does not claim systematic literature coverage, formal verification, penetration testing, or certification. A requirement is useful here only if an evaluator can identify both its enforcement point and evidence that could falsify it.

The unit of representation is a role and a validated decision class, not a personality. The unit of authorization is an interaction with a declared recipient set and purpose. Military, clinical, legal, credit, employment, and other consequential decisions require additional domain-specific assessment. The specification does not authorize autonomous use in any such domain.

3. Data admission and role memory

Before ingestion, each source needs an accountable owner, an authorized purpose, retention instructions, applicable restrictions, and an identity/permission mapping. Connector access is necessary but insufficient: a connector's technical ability to read a file does not establish permission to repurpose it for modeling historical decisions. Privileged, personal, legally restricted, or purpose-incompatible material must be excluded or handled under a separately approved workflow.

The proposed role memory distinguishes four record classes:

Record class	Minimum content	Permitted interpretation
Current policy	Owner, version, effective dates, authority	Governs only within its stated scope
Historical decision episode	Question, evidence then available, disposition, date, role, outcome if known	Contextual precedent, not current instruction

Record class	Minimum content	Permitted interpretation
Current role mandate	Incumbent, delegation limits, escalation route	Defines the present authority boundary
Generated synthesis	Source links, generation version, uncertainty	A revisable interpretation, not a new source fact

A decision episode should distinguish explicit rationale from inferred rationale. Missing reasons remain missing. Conflicting accounts remain visible; an extractor must not resolve disagreement merely by selecting the most fluent explanation. When outcomes are unknown, the record must not imply that the decision succeeded. These rules reduce the risk that historical records become unjustified lessons.

Role succession creates a deliberate discontinuity. The successor may change policy, objectives, risk tolerance, or permitted disclosure. Historical records therefore carry time boundaries and source-specific permissions. A former incumbent's personal notes do not become organization-wide memory merely because another person occupies the role.

4. Authorization model

For a candidate evidence item d, request q, requester u, audience A, purpose p, and time t, evidence may enter a recipient-visible response only if every applicable check succeeds. The proposed predicate is:

Allowed(d,q,u,A,p,t) = tenant match AND admissible purpose AND valid requester access AND valid audience access AND current policy permission AND acceptable permission freshness.

Every recipient of a common response must be authorized for its supporting content and permitted derived disclosures. If a mixed audience cannot receive the same answer, the assistant should produce an appropriately limited common answer or route separate authorized responses. It must not reveal a protected fact by saying that a secret document exists, through a citation title, or through a revealing refusal message.

Permission to retrieve evidence is separate from permission to perform an action. An employee's right to read a budget does not grant approval authority. An executive title does not by itself establish access to every record. Current tenant identity, role assignment, policy and source permissions govern access; conversational claims such as “the CEO said it is fine” are not credentials.

Enforcement belongs in trusted services around the model. The model may request an operation but cannot widen its own scope, supply authoritative ACLs, or approve its own proposal. Unknown identity, incomplete audience membership, expired delegation, unavailable policy checks, or stale permission state trigger a safe decline or a narrower response.

4.1 Microsoft integration example

Microsoft's September 17, 2026 documentation distinguishes generally available security-filter patterns from several native permission features marked preview, and describes source-specific permission synchronization [4]. This means that “uses existing Microsoft permissions” must be an implemented and tested integration claim, with connector, API version and synchronization behavior specified. It cannot be treated as automatic inheritance across all enterprise sources. The Trust Layer requirements below are the author's proposed controls, not a claim that Microsoft supplies them end to end.

Each connector needs a documented revocation pathway. The release record should specify its maximum accepted permission age and what happens when that bound is exceeded. Cached responses, embeddings, summaries, citation previews and exported transcripts require the same attention as raw documents. A revocation should invalidate affected derived artifacts or prevent their further use until reauthorization. Demonstrating an indexed ACL once does not establish continuous access correctness.

5. Response and authority states

Level	Proposed capability	Required authority boundary
L0: evidence	Retrieve and summarize admissible material	No recommendation or execution implied
L1: draft	Prepare a recommendation or message	Clearly labeled draft; source and uncertainty visible
L2: confirm	Submit a bounded proposal for human confirmation	Authorized human confirms the specific content, audience and action
L3: delegate	Execute a preauthorized, reversible low-consequence action	Explicit decision-class mandate, limits, expiry, monitoring and rollback

No level licenses an assistant to impersonate a person. Recommendations are attributed to the assistant and associated role, with the accountable human named where appropriate. High-consequence or professionally reserved acts remain outside delegation unless a separate lawful and validated mechanism explicitly authorizes them. Being able to generate an answer does not satisfy that condition.

A proposal moves through receipt, eligibility checking, evidence assembly, drafting, validation, confirmation if required, delivery or execution, and recording. Any failed check routes to safe decline or escalation. Before execution, the system rechecks identity, audience, policy, delegation and freshness; an earlier retrieval decision is insufficient. An idempotency key prevents duplicated actions after retries. A human approval is tied to a digest of the approved proposal so that later changes cannot silently

inherit approval.

6. Voice, meetings and messages

The proposed mobile experience begins with authenticated identity and a visible indication that the user is speaking with an AI assistant. An audio session should not use voice similarity as proof of authority. The assistant must distinguish a request for information, a request to brainstorm, and a request to commit the organization to action. Readback or an accessible confirmation interface should resolve ambiguous amounts, names and intended recipients before consequential delivery.

In a meeting, the audience can change mid-conversation. A newly joined participant changes what may be disclosed. A proposed control is to reassess the audience before each response and to pause protected content when attendance is uncertain. Recording and transcription follow the organization's approved notice, consent and retention policy; the specification assumes neither universal consent nor universal permission to record. A later meeting summary must be authorized for its own recipients.

A twin attending a meeting should identify itself, state its delegated scope, record requests requiring the incumbent, and avoid representing a generated opinion as a settled human commitment. It may provide supported context, identify conflicts, and prepare options. A request that exceeds its mandate becomes an escalation with the evidence packet needed by the human reviewer. This may reduce preparation time; it does not establish that a meeting can safely be replaced.

Drafting an email or Slack message, sending it, and reacting to a subsequent reply are separate operations. A draft may remain private to its reviewer. Sending requires a recipient and content check. A reply that introduces a new recipient or purpose restarts authorization. Quoted text and attached files are untrusted inputs, not instructions to change the assistant's controls.

7. Audit record and contestability

The proposed audit record records observable inputs, sources, decisions and actions rather than private chain-of-thought. A useful record includes a concise evidence-based explanation sufficient to understand the disposition, plus the following fields:

Field group	Required elements
Identity and scope	Tenant, pseudonymous request ID, requester and recipient references, role mandate, decision class, purpose
Time and policy	Request, readiness, permission-check and disposition timestamps; policy and delegation versions

Field group	Required elements
Evidence	Source and version IDs, authorized excerpts or controlled references, retrieval method, contradictions and exclusions
System	Model, prompt/configuration, retrieval, connector and action-gate versions
Disposition	Draft/decline/escalation/action; confidence limitations; reviewer and confirmation reference where required
Delivery	Intended and actual recipients, output digest, tool/action ID, result and retry linkage
Governance	Retention class, audit-access policy, incident reference, correction and appeal linkage

Auditability creates its own sensitive dataset. Raw payloads should not be copied into logs indiscriminately. Use controlled references, minimization, encryption and access separation; keep the evidence necessary for the approved audit purpose. Hashes can support integrity checks but do not prove that the original source was true. Append-only designs do not automatically settle conflicts among retention, correction, deletion and legal-hold obligations. Those conflicts need a documented governing decision.

A person affected by a recommendation needs a route to challenge the evidence, the role representation, or the decision itself. Corrections should propagate to affected summaries and identify outputs requiring reconsideration. The human reviewer must be able to override the system without silently erasing the original record. Worker telemetry should not become an undisclosed employee-performance score.

8. Failure modes and acceptance evidence

Failure mode	Proposed control	Evidence required before release
Cross-tenant or unauthorized disclosure	Isolation plus pre-generation and delivery authorization	Adversarial canary tests across sources, roles, caches and audiences
Revoked access still works	Revocation propagation and stale-state decline	Measured revoke-to-deny behavior for every supported connector
Historical precedent overrides new policy	Effective-date resolution and conflict handling	Superseded-policy cases with expected current-policy dispositions
Prompt injection causes an action	Separate policy gate and allowlisted tools	Malicious content cases; blocked actions and retained audit evidence
Plausible but unsupported rationale	Evidence-linked claims and missing-rationale labels	Blinded review of support, omissions and fabricated source links
Approval transferred to changed content	Proposal digest and reapproval	Tampered-recipient and changed-amount cases
Meeting participant gains restricted content	Audience refresh and bounded disclosure	Join/leave tests, transcript forwarding and citation-preview tests
Duplicate execution after retry	Idempotency, state tracking and rollback	Timeout, crash and retry fault-injection cases
Audit becomes a leakage route	Minimized records and separate audit permissions	Audit-access and export tests, including removed users

Passing a finite test set establishes only performance on that set. Report denominators, severity, test generation and coverage gaps. A release may pass one decision class and fail another. A global “trust score” should not conceal a critical authorization failure.

9. Evaluation and release gates

Evaluation should begin with synthetic and authorized retrospective cases, then progress to prospective shadow mode in which outputs do not influence operations. Assisted use requires explicit case-level eligibility and accountable review. WT-300-001 proposes how to measure time, quality, review burden and incidents [5]. No proposed gate is evidence that a gate has already been met.

Before any live pilot, the operator should identify: a named risk owner; permitted sources and decision classes; prohibited actions; permission-freshness bounds; a quality rubric

and acceptable error boundary; an incident procedure; and a continuity process if the assistant is unavailable. These are configuration choices to be justified by the use case, not universal numerical thresholds.

Model, corpus, prompt, policy and connector changes can invalidate prior evidence. A material change requires targeted revalidation of affected claims. Regression tests should include known failure cases and unseen holdouts, with test-set contamination recorded. Rollback needs both a technical procedure and a decision about whether outputs already delivered require correction.

10. Limitations and research agenda

The proposed architecture may increase latency because authorization and review take time. Its suitability therefore depends on whether any reduced waiting outweighs preparation and validation burden without worsening decision quality. Record completeness, interpretation fidelity and appropriate abstention remain open empirical questions. Historical precedent can encode bias even when provenance and permissions are correct.

The specification does not demonstrate Microsoft integration, meeting attendance, autonomous messaging, security certification or regulatory compliance in WisdomTwin. Independent implementation review, adversarial evaluation, usability work and domain-specific legal assessment remain necessary. The useful next result is a documented test outcome, including failures, rather than a generalized claim that enterprise judgment has been captured.

Declarations

Interest and authorship: Prepared for Roman Bodnarchuk, WisdomTwin co-founder and CEO, who has a financial interest in the proposed system. Generative AI assisted in synthesis, drafting and production. Named authorship does not imply independent technical validation.

Evidence and data: No new empirical data, enterprise records or human-participant research. Requirements and examples are conceptual. The author is responsible for the final text and any later submission.

Funding: No additional research-funding information was supplied for this companion draft.

Publication: Working paper, not peer reviewed; DOI unassigned.

References

1. Bodnarchuk, R. (2026). Judgment latency in regulated enterprises: A conceptual framework and research agenda for role-specific AI decision support. WT-100-001, v2.0. https://wisdomtwin.ai/research/wt-100-001

2. Tabassi, E. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1. https://doi.org/10.6028/NIST.AI.100-1

3. National Institute of Standards and Technology. (2024). Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile. NIST AI 600-1. https://doi.org/10.6028/NIST.AI.600-1

4. Microsoft. (2026, September 17). Document-level access control in Azure AI Search. Documentation accessed September 24, 2026. https://learn.microsoft.com/en-us/azure/search/search-document-level-access-overview

5. Bodnarchuk, R. (2026). Measuring judgment latency and net time recovery: A staged pilot protocol for role-specific AI decision support. WT-300-001, v1.0. Companion working paper; no DOI.