Quuipu

SERIE QUUIPU RESEARCH & ARCHITECTURE

WHITEPAPER 05

Trust Architecture

Provenance, explainability, controllo umano e intelligenza delimitata



Whitepaper fondativo · Italiano

Quuipu Edizione 2 · v1

Identità della pubblicazione

Pubblicazione
Trust Architecture

Serie
Serie Quuipu Research & Architecture

Numero di pubblicazione
Whitepaper 05

Edizione
Quuipu Edizione 2 · v1

Lingua
Italiano

Edizione storica
Cognisphere Edizione 1

Questa pubblicazione succede a Cognisphere Edizione 1. I PDF storici restano immutabili.

Indice

01. La fiducia come architettura

02. Trust Architecture

03. Provenance

04. Evidence

05. Confidence

06. Uncertainty

07. Explainability

08. Claimability

09. Human control

10. Fallback

11. Auditability

12. Bounded interaction

13. Policy architecture

14. Governance

15. Logging

16. Decision trace

17. Trust states

18. LLM integration

19. EU AI Act: transparency context

20. Transparency by design

21. Human oversight patterns

22. Trust in multi-tenant systems

23. Security and privacy

24. Testing trust

25. Trust metrics

26. Governance operating model

27. Trust invariants

28. La promessa finale

A. Decision Trace reference model

B. Trust failure matrix

C. EU transparency design checklist

D. Trust Architecture invariants

Preamble

Provenance, explainability, controllo umano e intelligenza delimitata

> Questa pubblicazione succede a Cognisphere Edizione 1. Quuipu Edizione 2 preserva la tesi storica aggiornando brand e identità di pubblicazione attivi.

Preamble

Provenance, Explainability, Human Control and Bounded Intelligence

FOUNDATIONAL PRINCIPLE

Fiducia senza cessione di controllo L'intelligenza adattiva diventa affidabile quando autorità della fonte, limiti del sistema e controllo umano restano visibili ed effettivi.

The experience may adapt. The source of truth does not.

Canonical Source ↓

Evidence & Provenance ↓

Bounded Runtime ↓

Policy & Claimability ↓

Decision Trace ↓

Human Control ↓

Adaptive Experience

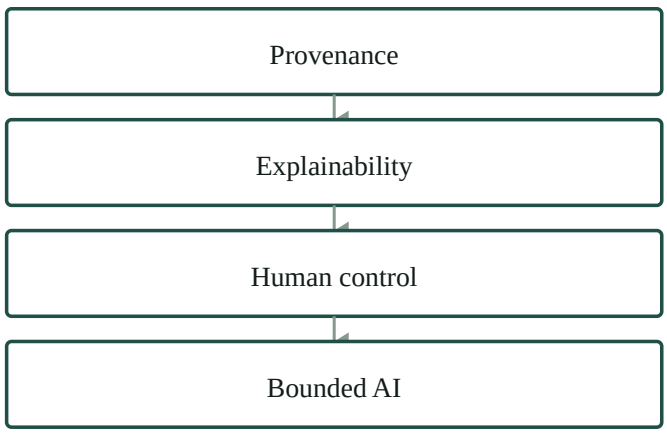


Figure 1 — Catena di trust.

TRUST MODEL

Quattro piani di fiducia Plane Primary responsibility Core artifacts

Source & Evidence Authority and support SourceRef, EvidenceRef, provenance

Runtime Decision Bounded resolution and policy Policy ID, reason code, integrity/trust state

Experience & Control Disclosure and user agency Source link, mode state, override, fallback

Governance & Audit Accountability and change control Logs, review gates, retention, incident trace

CAPITOLO 01

La fiducia come architettura

La fiducia in un'esperienza adattiva non può dipendere dall'impressione di intelligenza. Deve derivare da proprietà osservabili: origine dell'informazione, limiti del sistema, capacità di spiegare una trasformazione, controllo umano e possibilità di tornare alla fonte.

Trust is earned by structure Quuipu tratta la fiducia come risultato di contratti tecnici e di governance. Un output fluido non è più affidabile perché sembra convincente.

The invariant The experience may adapt. The source of truth does not. L'adattamento può modificare forma, percorso e granularità; non trasferisce l'autorità dalla fonte al runtime.

Trust surface L'interfaccia deve rendere percepibili almeno origine, stato, qualificazioni e possibilità di controllo, senza trasformare ogni interazione in un audit tecnico.

CAPITOLO 02

Trust Architecture

La Trust Architecture è il layer trasversale che collega runtime, esperienza e governance affinché ogni adattamento importante possa essere ricondotto a fonti, policy, decisioni e controlli.

Four planes Il modello distingue Source & Evidence Plane, Runtime Decision Plane, Experience & Control Plane e Governance & Audit Plane.

Separation of concerns Evidence stabilisce supporto; confidence descrive una valutazione interna; explainability ricostruisce il perché; governance stabilisce ciò che è consentito. Nessuno di questi concetti sostituisce gli altri.

Bounded intelligence L'intelligenza è utile quando opera entro scope, capability, policy e source authority espliciti.

Source & Evidence Plane ↓

Runtime Decision Plane ↓

Experience & Control Plane ↓

Governance & Audit Plane

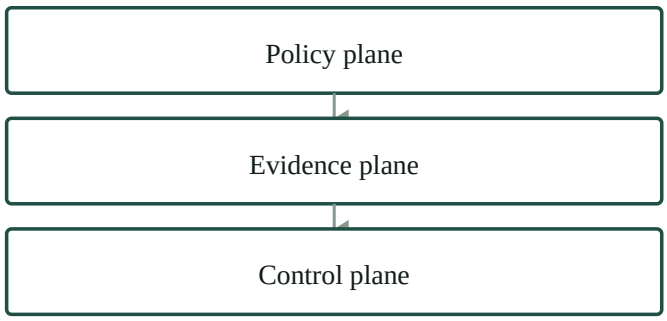


Figure 2 — Piani di Trust Architecture.

CAPITOLO 03

Provenance

La provenance collega ciò che l'utente vede a ciò da cui deriva. È una condizione di fiducia, non un ornamento bibliografico.

Source provenance Identifica fonte, versione, locator e authority scope.

Transformation provenance Registra quale capability o trasformazione ha prodotto la rappresentazione.

Decision provenance Collega source selection, policy, uncertainty e fallback alla decisione di emissione.

Progressive disclosure La UI può mostrare un source link semplice, mentre audit e debugging conservano un trace più ricco.

CAPITOLO 04

Evidence

Evidence è il supporto ammissibile per claim e trasformazioni. Il suo ruolo è impedire che plausibilità, memoria del modello o somiglianza semantica vengano confuse con prova.

Evidence units Passaggi, record, campi strutturati o relazioni possono diventare unità di evidence se conservano origine e scope.

Coverage Un claim materiale richiede copertura sufficiente, non soltanto una fonte genericamente pertinente.

Strength La forza dell'evidence dipende da authority, directness, version e policy di dominio.

Evidence gaps Un gap deve produrre qualification, richiesta di contesto, riduzione dello scope o fallback.

CAPITOLO 05

Confidence

Confidence è un segnale operativo, non una misura universale di verità. Deve essere interpretata nel contesto del tipo di decisione e delle evidenze disponibili.

Confidence is not authority Un modello può essere molto confidente e comunque non avere diritto di formulare un claim.

Calibrated use Confidence può contribuire a routing, escalation o scelta del fallback, ma non deve essere l'unico gate.

User projection Quando mostrata all'utente, una confidence numerica può creare falsa precisione. Spesso stati qualificativi sono più comprensibili.

Calibration Le soglie devono essere validate per dominio e materialità, non copiate fra use case diversi.

Signal Useful for Not sufficient for

Confidence Routing / escalation Truth or authority

Evidence strength Claim support Policy permission

Integrity state Semantic safety User disclosure by itself

Human review Additional assurance Permanent correctness

CAPITOLO 06

Uncertainty

Quuipu deve poter rappresentare incertezza di fonte, resolution, trasformazione e policy senza nasconderla dietro linguaggio assertivo.

Known unknowns Ambiguità, dati mancanti e conflitti possono essere esplicitamente modellati.

Uncertainty propagation Se una premessa è incerta, le rappresentazioni derivate non devono perdere tale qualificazione.

User handling Il sistema può mostrare alternative, chiedere chiarimento, ridurre scope o tornare a Original.

Governance threshold Domini ad alta materialità richiedono soglie più conservative.

CAPITOLO 07

Explainability

Explainability significa fornire una spiegazione utile e proporzionata di come il sistema è arrivato a una rappresentazione o decisione.

Why this information Quali segnali di contesto e quali fonti hanno reso un contenuto rilevante.

Why this representation Perché è stata scelta una modalità Simple, Step-by-step, Explore o altra composizione.

Why not Perché una capability è stata rifiutata, qualificata o degradata.

No fake rationale Una spiegazione non deve essere una narrazione post-hoc inventata dal modello; deve derivare da trace e decision data effettivi.

CAPITOLO 08

Claimability

Claimability definisce ciò che il sistema è autorizzato ad affermare. È il ponte fra evidence, policy e user-facing language.

Canonical claim Affermazione direttamente attribuibile alla fonte autorevole.

Supported derivation Conclusione sostenuta dalle fonti entro una regola ammessa.

Qualified inference Inferenza utile ma condizionale, che richiede qualificazione.

Unsupported claim Non deve essere emesso come trusted output.

Materiality Più il claim incide su diritti, denaro, sicurezza, obblighi o decisioni, più alta deve essere la soglia.

Claim class Trust treatment Typical action

Canonical Highest source attribution Show source

Supported derivation Evidence-backed Show provenance

Qualified inference Conditional Disclose qualification

Unsupported Below threshold Block / fallback

CAPITOLO 09

Human control

Human control non significa inserire una persona in ogni ciclo. Significa progettare punti reali di intervento, correzione e responsabilità.

User override L'utente può cambiare intent, modalità, fonte o tornare all'esperienza standard.

Operator override Un operatore autorizzato può disabilitare capability, correggere policy, escludere fonti o forzare fallback.

Review gates Determinati claim o domini possono richiedere revisione umana prima della pubblicazione o dell'azione.

No dark autonomy Il sistema non deve espandere autonomamente il proprio mandato oltre le capability autorizzate.

CAPITOLO 10

Fallback

Fallback è una proprietà di fiducia: dimostra che Quuipu preferisce un'esperienza meno adattiva a una risposta non sufficientemente supportata.

Technical fallback Runtime indisponibile, timeout o errore: ritorno alla superficie canonica.

Semantic fallback Runtime disponibile ma integrity insufficiente: Original, scope ridotto, alternative o refusal.

Policy fallback Una policy vieta una trasformazione o richiede human review.

Graceful degradation La degradazione deve essere comprensibile e non interrompere il percorso fondamentale dell'utente.

Normal adaptive experience ↓

Qualification ↓

Reduced scope ↓

Return to Original ↓

Human review ↓

Refusal

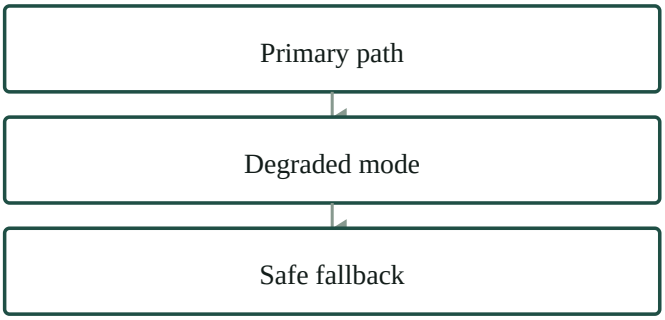


Figure 3 — Scala di fallback.

CAPITOLO 11

Auditability

Auditability è la capacità di ricostruire eventi rilevanti senza trasformare il prodotto in un sistema di sorveglianza.

What to reconstruct Input di contesto rilevante, source set, policy, capability, decision state, output reference e fallback.

Minimality Il trace deve raccogliere ciò che serve a sicurezza, qualità e accountability, evitando dati non necessari.

Immutability and retention Per eventi ad alto valore, integrità del log e retention policy devono essere definite.

Audit views Debugging tecnico, audit di governance e spiegazione utente hanno granularità differenti.

CAPITOLO 12

Bounded interaction

Quuipu non assume che più libertà conversazionale equivalga a migliore esperienza. L'interazione può essere limitata per preservare authority, sicurezza e comprensibilità.

Declared intent L'utente dichiara ciò che cerca di capire senza essere profilato automaticamente.

Bounded topics Explore attraversa relazioni governate anziché generare percorsi illimitati.

Capability boundaries Ogni capability espone input, output, policy e failure modes definiti.

Context boundary Il sistema usa il contesto necessario allo scopo corrente, non un'identità comportamentale illimitata.

CAPITOLO 13

Policy architecture

Le policy trasformano principi di trust in regole operative versionabili.

Source policy Quali fonti sono ammesse e con quale authority.

Transformation policy Quali trasformazioni sono consentite per classe di contenuto.

Claim policy Quali claim richiedono evidence primaria, qualificazione, doppio controllo o refusal.

Interaction policy Quali modalità e capability sono disponibili in uno scope.

Retention policy Quali trace e state possono essere conservati e per quanto tempo.

CAPITOLO 14

Governance

Governance assegna responsabilità e rende modificabili in modo controllato le regole che delimitano l'intelligenza.

Ownership Source owners, product owners, security, legal/compliance e engineering hanno responsabilità diverse.

Change control Modifiche a fonti, modelli, prompt, policy o capability possono richiedere review e release gate.

Exception process Le eccezioni devono essere esplicite, temporanee quando possibile e auditabili.

Governance as runtime input Le decisioni di governance non restano nei documenti: diventano configurazione e contratti eseguibili.

CAPITOLO 15

Logging

Il logging di Quuipu deve separare osservabilità tecnica, decision trace e behavioral analytics.

Operational logs Errori, latenza, timeout, capability health e fallback tecnico.

Decision logs Policy applicata, source class, integrity state, reason code e human override.

Behavioral analytics Sono una categoria separata e opzionale; non sono un prerequisito per la Trust Architecture.

Privacy by design No telemetry/no persistence può essere una configurazione valida per il cognitive context, mantenendo health logging minimizzato.

CAPITOLO 16

Decision trace

Il decision trace è la sequenza strutturata che spiega come il runtime è passato da contesto e fonti a un'esperienza emessa.

Trace, not chain-of-thought Il trace contiene decision data osservabili - regole, source selection, state transitions e reason codes - non ragionamento interno non necessario.

Deterministic anchors Policy ID, source version, capability version e integrity state rendono il trace verificabile.

Branching Il trace deve registrare perché una strada è stata scelta, qualificata o scartata.

Replayability Dove richiesto, un trace può consentire di ricostruire le condizioni di una decisione senza pretendere identità perfetta di un modello probabilistico.

Trace anchor Example Purpose

Context declaredIntent + mode Why the request was resolved this way

Source sourceId + version What authority was used

Policy policyId + version What rule bounded the decision

Capability capability + version What transformation acted

Decision reasonCode + state Why emitted/qualified/blocked

Override actor + action What human control changed

CAPITOLO 17

Trust states

Un'esperienza può esporre stati di trust distinti dallo stato tecnico del runtime.

Trusted Source, evidence e policy soddisfano la soglia prevista.

Qualified L'esperienza è utile ma presenta uncertainty o inferenze che richiedono disclosure.

Human-reviewed Un operatore ha validato il risultato o la classe di risultato.

Degraded La capability cognitiva è ridotta e il sistema utilizza una superficie più sicura.

Blocked Policy o integrità impediscono l'emissione.

CAPITOLO 18

LLM integration

Se Quuipu usa LLM, il modello resta una capability subordinata alla Trust Architecture.

Provider abstraction Il runtime deve poter cambiare modello senza perdere provenance, policy, fallback e trace.

Prompt is not governance Un'istruzione nel prompt non sostituisce enforcement architetturale.

Generated output as candidate L'output del modello viene trattato come candidato da validare, non come fonte autorevole.

Disclosure Quando l'interazione o il contenuto ricade in obblighi di trasparenza applicabili, la disclosure deve essere progettata come parte dell'esperienza.

CAPITOLO 19

EU AI Act: transparency context

Dal 2 agosto 2026 l'articolo 50 dell'AI Act si applica a determinate categorie di sistemi e contenuti AI. La Commissione europea ha pubblicato il 20 luglio 2026 linee guida dedicate per chiarirne l'ambito e l'applicazione. Questa sezione fornisce contesto progettuale, non consulenza legale.

Interactive AI Per sistemi che interagiscono direttamente con persone, l'articolo 50(1) prevede, nei casi applicabili, che le persone siano informate dell'interazione con un sistema AI, salvo che ciò sia ovvio.

Synthetic content L'articolo 50(2) riguarda, nei casi applicabili, marking machine-readable di contenuti sintetici o manipolati generati da AI; esistono scope, eccezioni e regole transitorie specifiche.

Deployers Ulteriori obblighi riguardano, tra l'altro, deepfake e determinate pubblicazioni testuali su temi di interesse pubblico senza human review/editorial control.

Architecture implication Per Quuipu, disclosure, provenance, content marking, logging e human control devono essere capability configurabili e non aggiunte retroattive.

Legal boundary L'applicabilità concreta dipende dal ruolo di provider/deployer, dal tipo di sistema e dal contesto d'uso; va valutata caso per caso.

AI Act transparency area Architecture response Boundary

Direct AI interaction Visible AI invocation/disclosure Only where legally applicable or product-appropriate

Synthetic content marking Emission metadata/marking capability Depends on content/system scope and exceptions

Deepfake/public-interest disclosure Deploy-time disclosure controls Role and context specific

Compliance evidence Versioned policy + trace + logs Not a substitute for legal assessment

CAPITOLO 20

Transparency by design

La trasparenza più efficace è incorporata nei contratti di esperienza prima del deployment.

AI interaction disclosure L'utente deve poter capire quando una capability AI è effettivamente invocata, ove richiesto o appropriato.

Derived-content disclosure Una rappresentazione derivata deve essere distinguibile dal contenuto canonico.

Machine-readable metadata Quando richiesto, il layer di emissione può associare marker o metadata senza alterare la fonte canonica.

First exposure Le disclosure user-facing devono comparire nel momento utile, non essere nascoste esclusivamente in policy o footer.

CAPITOLO 21

Human oversight patterns

Diversi domini richiedono diversi pattern di oversight.

Human-on-request L'utente può escalare o chiedere verifica.

Human-on-exception Il sistema coinvolge un operatore per conflitti, bassa evidence o claim ad alta materialità.

Human-before-publish Output destinati a pubblicazione esterna possono richiedere approvazione.

Human-in-governance Esperti definiscono source authority, policy e test anche quando non intervengono su ogni singola interazione.

Oversight pattern Trigger Latency

On request User asks for verification Asynchronous possible

On exception Conflict / low evidence Before trusted completion

Before publish External/high-impact publication Approval gate

In governance Policy/source/model changes Release cycle

CAPITOLO 22

Trust in multi-tenant systems

In un deployment multi-tenant, trust richiede isolamento non soltanto dei dati ma anche di authority, policy, trace e controllo.

Tenant policy Ogni tenant può avere regole di fonte, claimability e retention differenti.

Trace isolation Audit e decision trace non devono attraversare confini tenant.

Shared capabilities Modelli e core possono essere condivisi, mentre context, evidence e governance restano isolati.

Delegated administration Ruoli autorizzati possono gestire policy senza acquisire accesso indiscriminato al contenuto di altri tenant.

CAPITOLO 23

Security and privacy

Trust Architecture e security architecture si intersecano ma non coincidono.

Least privilege Host, runtime e capability accedono soltanto a fonti e contesti necessari.

Credential boundary Credenziali grezze non devono diventare parte del cognitive context.

Sensitive traces Decision trace e log possono contenere metadati sensibili e richiedono access control, retention e minimizzazione.

No trust without security Provenance e explainability perdono valore se source identity o log possono essere alterati senza controllo.

CAPITOLO 24

Testing trust

La fiducia deve essere verificata attraverso failure injection, policy tests, provenance tests e scenari umani.

Policy tests Verificano che claim o capability vietati vengano effettivamente bloccati.

Trace tests Verificano completezza e coerenza dei reason codes e degli anchor.

Fallback tests Simulano timeout, missing source, conflitto, uncertainty e operator override.

Disclosure tests Verificano che stati e disclosure siano percepibili, accessibili e tempestivi.

Red-team trust Test avversariali cercano authority confusion, prompt bypass, cross-tenant leakage e fabricated explanations.

CAPITOLO 25

Trust metrics

Le metriche devono misurare se il sistema merita fiducia, non soltanto se viene percepito come affidabile.

Provenance completeness Percentuale di output derivati con source path sufficiente.

Unsupported claim rate Frequenza di claim emessi senza evidence adeguata.

Correct fallback rate Quanto spesso il sistema degrada quando dovrebbe.

Override effectiveness Capacità di human override di produrre un cambiamento reale e tracciabile.

Disclosure coverage Copertura delle situazioni in cui AI interaction o derived content devono essere resi visibili.

Metric What it tests Failure signal

Provenance completeness Traceability Derived output without source path

Unsupported claim rate Evidence discipline Claims beyond support

Correct fallback rate Conservatism Unsafe continuation

Override effectiveness Human control Override has no effect

Disclosure coverage Transparency AI/derived state hidden

CAPITOLO 26

Governance operating model

Una Trust Architecture diventa operativa quando ruoli, review e incident process sono chiari.

Trust owner Mantiene invarianti, risk taxonomy e metriche.

Domain authority Definisce source authority e materiality nel dominio.

Engineering Implementa enforcement, logging, isolation e tests.

Compliance/legal Interpreta obblighi applicabili e approva policy di disclosure dove necessario.

Incident review Analizza trust failures e produce remediation verificabile.

CAPITOLO 27

Trust invariants

Gli invarianti sono le regole che devono sopravvivere a cambi di modello, host e deployment.

Source truth is stable L'adattamento non cambia silenziosamente la fonte autorevole.

Evidence precedes claim I claim materiali richiedono supporto.

Uncertainty remains uncertainty La UI non trasforma incertezza in certezza.

Control is effective Override e fallback devono cambiare davvero il comportamento.

Trace is factual Explainability deriva da eventi reali, non da razionalizzazioni generate.

Governance is enforceable Le policy sono applicate dal runtime, non soltanto documentate.

CAPITOLO 28

La promessa finale

Quuipu Trust Architecture mira a rendere l'intelligenza adattiva utilizzabile proprio perché ne delimita il potere. Il sistema può aiutare di più quando sa mostrare da dove viene l'informazione, cosa ha fatto, quanto può sostenere e quando deve fermarsi.

Adaptive experience La forma può rispondere al contesto.

Stable authority La fonte autorevole mantiene il proprio ruolo.

Visible limits Uncertainty, fallback e policy diventano parti leggibili dell'esperienza.

Human sovereignty La persona e l'organizzazione conservano punti reali di controllo.

The principle The experience may adapt. The source of truth does not.

CAPITOLO A

Decision Trace reference model

Field Example Trust purpose

traceId immutable event ID Correlation

contextRef session-scoped snapshot Resolution context

sourceRefs IDs + versions Authority

evidenceRefs locators + strength Support

policyRef policy/version Governance

capabilityRef mode/model/version Transformation

decision emit/qualify/fallback/block Outcome

reasonCodes structured codes Explainability

overrideRef actor/action/time Human control

CAPITOLO B

Trust failure matrix

Failure Risk Required response

Missing provenance Untraceable output Qualify or block

High confidence / low evidence False certainty Evidence gate wins

Policy bypass Unbounded intelligence Block + incident trace

Fabricated explanation False transparency Use factual reason codes only

Override ignored Loss of human control Fail safe + alert

Cross-tenant trace Privacy/authority leakage Reject + isolate

Hidden AI invocation Transparency failure Disclosure remediation

Stale source Wrong authority state Invalidate / refresh

CAPITOLO C

EU transparency design checklist

This checklist translates current EU transparency context into architecture questions. It is not a legal compliance checklist.

Question Architecture hook Owner

Does the user directly interact with an AI capability? Invocation state + disclosure surface Product / legal

Can the system generate or manipulate synthetic content? Marking metadata at emission Engineering / compliance

Is derived content externally published? Deploy-time disclosure policy Publisher/deployer

Is human review/editorial control present? Review state + decision trace Governance

Can compliance measures be evidenced? Versioned policy, logs, trace Compliance / audit

Are pre-2 Aug 2026 systems relevant? Transition/version inventory Legal / operations

CAPITOLO D

Trust Architecture invariants

TA-01 · Stable source of truth Adaptation never silently redefines canonical authority.

TA-02 · Evidence before material claim Support is required before assertion.

TA-03 · Confidence is not authority Confidence never overrides evidence or policy.

TA-04 · Uncertainty is preserved Qualification propagates into the experience.

TA-05 · Explainability is factual Explanations derive from real decision data.

TA-06 · Human control is effective Override changes runtime behavior and is traceable.

TA-07 · Fallback is first-class Safer degradation exists for technical, semantic and policy failure.

TA-08 · Intelligence is bounded Capabilities operate inside explicit scope and policy.

TA-09 · Governance is executable Policy is enforced by runtime contracts.

TA-10 · Logging is minimized Auditability does not require unlimited behavioral collection.

TA-11 · Tenant trust is isolated Authority, policy and trace remain tenant-scoped.

TA-12 · Transparency is designed in Applicable disclosure is an experience capability, not an afterthought.

APPENDIX E

Relationship to Semantic Integrity Semantic Integrity (WP04) Trust Architecture (WP05) Combined property

Preserves meaning Makes preservation visible/governable Trustworthy adaptation

Evidence & claimability Policy & decision trace Defensible claims

Integrity states Trust states + human control Safe experience behavior

Semantic fallback Technical/policy/human fallback Layered resilience

Source authority Governance ownership Stable source of truth

Provenance graph Auditability/explainability Reconstructable decisions

REFERENCES

Regulatory references European Commission, Guidelines on transparency obligations for providers and deployers of AI systems, published 20 July 2026. The Commission states that Article 50 transparency obligations apply from 2 August 2026. European Commission, Transparency obligations under Article 50 of the AI Act - Questions & Answers, 2026. This resource clarifies provider/deployer duties, direct AI interaction, synthetic-content marking, exceptions and transition rules. European Commission, Code of Practice on Transparency of AI-generated Content, 2026. The voluntary Code supports compliance with marking and labelling obligations under Article 50(2), (4) and (5).

Legal note The regulatory discussion in this whitepaper is architecture-oriented and does not constitute legal advice. Applicability should be assessed against the final legal text, Commission guidance, organizational role and deployment context.

Trust Architecture Provenance, Explainability, Human Control and Bounded Intelligence

THE EXPERIENCE MAY ADAPT. THE SOURCE OF TRUTH DOES NOT.