

Agentic AI Observability and Auditability Architecture

A Comprehensive Enterprise Architecture for Tracing, Monitoring, Explaining, Auditing, and Governing Autonomous AI Agent Behavior

Sanjeeve Kumar Gajadi*

*(HCL Technologies, Bengaluru, Karnataka, India)

Email: sanjeevekumar.g@hcltech.com

Abstract:

Agentic AI systems can reason, plan, invoke tools, delegate tasks, access enterprise data, and perform actions across applications with limited human intervention. These characteristics create observability and auditability challenges that are not fully addressed by traditional application logging or model monitoring. This paper proposes an Agentic AI Observability and Auditability Architecture that captures agent identity, objectives, planning steps, prompts, model versions, retrieved context, tool calls, policy decisions, approvals, execution outcomes, exceptions, and human interventions. The architecture introduces event schemas, trace correlation, tamper-resistant audit records, risk-aware telemetry, privacy controls, behavioral baselines, anomaly detection, and forensic reconstruction. It also defines architectural layers for telemetry generation, collection, normalization, policy evaluation, storage, analytics, dashboards, alerting, and audit evidence. The proposed design enables enterprises to operate autonomous and multi-agent systems with stronger transparency, accountability, incident response, regulatory readiness, and operational control.

Keywords — Agentic AI, AI Observability, AI Auditability, Autonomous Agents, Multi-Agent Systems, AI Governance, Traceability, AI Security, Audit Logging, Enterprise Architecture

I. INTRODUCTION

Traditional observability focuses on infrastructure, application performance, logs, metrics, and traces. Agentic AI introduces additional dimensions because behavior emerges from prompts, model reasoning, retrieved context, tool availability, delegation, and policy constraints.

Enterprises need observability that explains not only whether an agent failed, but also what it attempted, why a tool was selected, what data was accessed, which controls were applied, and what business outcome resulted.

II. ARCHITECTURE OBJECTIVES

The architecture must support real-time operations, post-incident forensics, compliance evidence, performance management, cost monitoring, model evaluation, and business accountability.

It should preserve sufficient context to reconstruct important decisions without exposing

unnecessary sensitive information or internal secrets.

III. TELEMETRY MODEL

Core telemetry includes agent identity, session identifier, objective, plan, model endpoint, model version, prompts, retrieved sources, tool calls, parameters, authorization decisions, outputs, confidence indicators, errors, and human interventions.

Events should use stable schemas and correlation identifiers so that multi-step and multi-agent workflows can be reconstructed end to end.

IV. TRACE AND EVENT ARCHITECTURE

Every significant agent action should generate a trace event containing timestamp, actor, action, target, policy result, execution state, and outcome.

Distributed tracing should connect agent activity with APIs, databases, workflows, integration platforms, and external services.

V. AUDIT RECORD ARCHITECTURE

Audit records should be append-only, time-synchronized, access-controlled, and retained according to regulatory and business requirements.

Cryptographic integrity controls, immutable storage, and evidence hashing can improve confidence that critical records were not altered after execution.

VI. POLICY DECISION OBSERVABILITY

Authorization, data access, tool permissions, transaction limits, content filters, and human-approval checks should emit explicit policy-decision events.

This enables auditors to determine whether controls were evaluated and whether an agent acted within its assigned authority.

VII. BEHAVIORAL MONITORING

Behavioral baselines can identify unexpected tool sequences, unusual data access, excessive retries, abnormal delegation, privilege escalation attempts, and unexplained changes in execution patterns.

Anomaly detection should be interpreted in business context because some rare actions may be legitimate while repetitive low-level anomalies can indicate emerging risk.

VIII. PRIVACY AND DATA MINIMIZATION

Observability must avoid creating a secondary sensitive-data repository. Logs should minimize payload capture, tokenize identifiers where appropriate, and apply role-based access controls.

Retention should reflect investigation needs, legal requirements, model assurance obligations, and data protection principles.

IX. HUMAN OVERSIGHT AND REVIEW

Human approvals, overrides, escalations, and manual corrections should appear within the same trace as agent actions.

This provides evidence that high-risk workflows received meaningful supervision and allows organizations to analyze when human intervention improves outcomes.

X. OPERATIONAL DASHBOARDS AND ALERTS

Dashboards should present success rate, latency, cost, token or compute usage, policy denials, tool

failures, exception rates, intervention frequency, and business outcomes.

Alerts should prioritize events based on impact, risk tier, autonomy level, sensitive data exposure, and ability to reverse the action.

XI. AUDIT AND FORENSIC RECONSTRUCTION

Forensic analysis should reconstruct the complete chain from initial request through planning, retrieval, tool execution, policy checks, approvals, and final action.

Evidence packages should support internal audit, external audit, security investigations, regulatory inquiries, and model-risk reviews.

XII. RESILIENCE AND FAIL-SAFE CONTROLS

Observability components should remain available during degraded operation because loss of telemetry can itself be a critical risk.

High-risk agents should fail safely when audit logging, policy enforcement, identity verification, or required monitoring is unavailable.

XIII. CONCLUSION

Agentic AI observability must extend beyond infrastructure monitoring to capture the full behavioral and governance context of autonomous execution.

The proposed architecture enables enterprises to trace, explain, audit, and control AI agents while supporting operational resilience, security, accountability, and compliance.

ACKNOWLEDGMENT

The author acknowledges the contributions of enterprise architecture practitioners, AI governance communities, security professionals, standards bodies, and technology researchers whose published work has informed the concepts synthesized in this paper.

REFERENCES

- [1] National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework (AI RMF 1.0), 2023.
- [2] National Institute of Standards and Technology, AI RMF Generative Artificial Intelligence Profile, 2024.

- [3] ISO/IEC 42001:2023, Information technology — Artificial intelligence — Management system.
- [4] ISO 31000:2018, Risk management — Guidelines.
- [5] ISO/IEC 27001:2022, Information security, cybersecurity and privacy protection — Information security management systems.
- [6] National Institute of Standards and Technology, SP 800-207: Zero Trust Architecture, 2020.
- [7] The Open Group, TOGAF Standard, 10th Edition, 2022.
- [8] ISACA, COBIT 2019 Framework: Governance and Management Objectives, 2019.
- [9] OWASP Foundation, OWASP Top 10 for Large Language Model Applications.
- [10] MITRE, ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems.
- [11] OECD, Recommendation of the Council on Artificial Intelligence.
- [12] IEEE, Ethically Aligned Design: A Vision for Prioritizing Human Well-being with Autonomous and Intelligent Systems.