

# Semantic Tracing in LLM-Based Multi-Agent Systems Using LangChain, LangGraph, and LangSmith for AI Governance

Audrey Rahimi

[arahimi@cougarnet.uh.edu](mailto:arahimi@cougarnet.uh.edu)

University of Houston <https://orcid.org/0000-0001-7501-7809>

---

## Research Article

**Keywords:** LLM-based multi-agent systems, agentic AI, LangChain, LangGraph, LangSmith, semantic tracing, AI governance, multi-agent communication, observability, instruction drift, hallucination, cybersecurity, Shadow AI, NIST AI RMF

**Posted Date:** July 28th, 2026

**DOI:** <https://doi.org/10.21203/rs.3.rs-10485157/v1>

**License:** © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.  
[Read Full License](#)

**Additional Declarations:** The authors declare no competing interests.

---

# Semantic Tracing in LLM-Based Multi-Agent Systems Using LangChain, LangGraph, and LangSmith for AI Governance

Audrey Rah <sup>1</sup> 

<sup>1</sup> Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204, USA

\* Correspondence: [arahimi@uh.edu](mailto:arahimi@uh.edu)

## Abstract

Autonomous multi-agent systems built on large language models exchange goals, constraints, evidence commitments, and tool-mediated actions through natural-language handoffs. Across hops, semantic meaning may drift, constraints may weaken, unsupported claims may propagate, and organizations may lose visibility into how decisions were produced. This paper develops a semantic-tracing framework that treats meaning preservation as a first-class measurand and operationalizes it with three LangChain ecosystem components that are genuinely used in the local experimental artifact: LangChain for LLM invocation, prompt templates, tool interfaces, structured-output parsing, and retries; LangGraph for stateful multi-agent orchestration (nodes, edges, workflow state, routing, and human-review control); and LangSmith for private tracing, span capture, metadata, and evaluation-linked observability. The system separates framework orchestration, LLM reasoning, deterministic safeguards, trace capture, semantic evaluation, and human decision. The framework is anchored in SHADOWAI-RISK, a live academic research prototype for enterprise AI governance and risk intelligence. In addition to the live prototype, this study implements a separate local experimental extension with (i) a preserved deterministic baseline pathway and (ii) a genuine LLM multi-agent condition (Inventory → Evidence → Governance → Human Review Gate) implemented with LangChain (ChatOpenAI, ChatPromptTemplate, JsonOutputParser, tool wrappers, retries), LangGraph (StateGraph nodes/edges/state), optional private LangSmith instrumentation, and a rule-based semantic evaluation layer. The full experimental matrix is complete at 450/450 (complete:true; deterministic 32, llm\_no\_eval 160, llm\_langsmith\_semantic\_eval 160, variability 50, ablations 48), with 0 terminal failures. Primary eval versus no-eval contrasts ( $n=160$  each) show Holm-corrected Mann–Whitney effects including lower TCS with evaluation (0.624 vs. 0.912, Holm-adjusted  $p < 0.001$ , Cliff  $\delta \approx -0.33$  medium), significantly lower WRS\_equal and WRS\_expert (small effects), non-significant SPS/AHF/IDR/TUA/GCR/HPR/HEP, higher latency with evaluation ( $\sim 126.2$  s vs.  $\sim 101.7$  s, Cliff  $\delta \approx 0.67$  large), non-significant token usage, and \$0 local cost; decision-family correctness did not differ (59/160 vs. 57/160; Fisher/ $\chi^2$  n.s.). Human calibration, inter-rater agreement, surveys, and production integration remain pending.

**Keywords:** LLM-based multi-agent systems; agentic AI; LangChain; LangGraph; LangSmith; semantic tracing; AI governance; multi-agent communication; observability; instruction drift; hallucination; cybersecurity; Shadow AI; NIST AI RMF

Case-study software artifact: SHADOWAI-RISK (<https://shadowai-risk.vercel.app/>)

## 1. Introduction

Organizations are currently negotiating whether and how to adopt AI under conditions of cybersecurity exposure, privacy risk, hallucination, accountability gaps, unauthorized AI use (Shadow AI), and potential loss of managerial control. Professional and regulatory institutions—including NIST, ISACA, European institutions, ENISA, and the OECD—are shaping expectations for trustworthy AI and organizational oversight [1–4].

At the same time, LLM-based systems are moving from single-agent assistants toward multi-agent workflows in which specialized agents plan, retrieve evidence, invoke tools, critique outputs, and hand work to other agents [5–8]. Classical multi-agent theory emphasized autonomy and coordinated interaction [9]. In LLM systems, coordination is mediated by natural-language messages whose meaning is brittle [10,11].

Observability platforms such as LangSmith capture end-to-end execution traces [12]. AgentOps research likewise argues that lifecycle observability supports safer agent operation [13]. Tracing alone, however, does not answer the governance question that matters in enterprise settings: whether agents preserved the meaning of what they were asked to do, under which constraints, on which evidence, and with what residual risk.

This paper addresses that problem by defining semantic tracing and binding it to enterprise AI risk workflows. Figure 1 summarizes the end-to-end research logic.

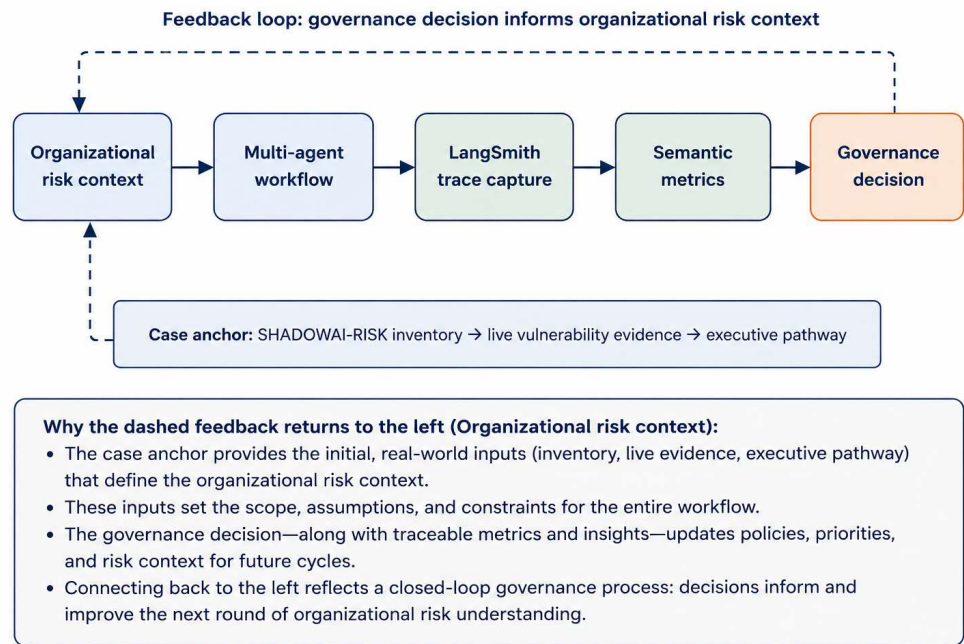


**Figure 1.** End-to-end research logic from organizational risk context through multi-agent workflow, LangSmith trace capture, semantic metrics, and governance decision.

## 2. Research Motivation and Gap

### 2.1 Motivation

Enterprise AI adoption debates require decisions to permit, restrict, stage, or continuously monitor AI tools. Shadow AI creates a visibility problem when AI-enabled work occurs outside approved inventories and logging regimes [14,15]. ENISA’s threat-landscape work positions AI as a material cybersecurity concern.

The NIST AI Risk Management Framework organizes trustworthy AI management around Govern, Map, Measure, and Manage [1]. The NIST Generative AI Profile highlights risks such as confabulation and misuse [16]. The EU Artificial Intelligence Act establishes

risk-based obligations and human-oversight expectations [2,17]. ISACA materials emphasize lifecycle governance and assurance [18,19]. These sources motivate systems that can show what agents did, what they communicated, and where controls should intervene.

SHADOWAI-RISK operationalizes part of this motivation as a live academic research prototype that connects an AI inventory to authoritative vulnerability intelligence and an executive decision pathway [20]. Semantic tracing is motivated as the complementary capability for dynamic observation of agent-mediated decision processes.

## 2.2 Research gap

Related work provides building blocks but leaves a gap at the intersection of semantic communication fidelity, tracing infrastructure, and enterprise governance decision support [13,21].

Table 1 synthesizes the research gap against major related-work strands.

**Table 1.** Research-gap synthesis against major related-work strands.

| Strand                  | What it covers                                | Unresolved for this study                       |
|-------------------------|-----------------------------------------------|-------------------------------------------------|
| LLM multi-agent surveys | Communication patterns, workflows, challenges | Governance metricization of semantic drift      |
| Agent surveys           | Architecture, tools, evaluation overview      | Enterprise control mapping and handoff fidelity |
| Hallucination surveys   | Unsupported generation measurement/mitigation | Propagation across agent hops                   |
| AgentOps / LangSmith    | Observability artifacts and tracing           | Semantic preservation scoring for governance    |
| NIST/ISACA/EU/ENISA     | Risk, assurance, regulatory expectations      | Operational multi-agent semantic tracing method |

The gap is therefore precise: there is no integrated, reproducible framework that (i) treats meaning preservation across multi-agent hops as a first-class measurand, (ii) binds those measurements to LangSmith-style traces, (iii) maps metric failures to governance controls, and (iv) demonstrates the approach in an enterprise AI risk application context while separating implemented features from proposed architecture.

## 3. Research Questions and Contributions

Table 2 presents the research questions.

Contributions are summarized in Figure 29 and linked to gap, method, output, and validation status.

The study contributes: (C1) a conceptual semantic-tracing construct and handoff-packet representation; (C2) a LangSmith-centered observability pipeline and formal metric suite (pending human calibration on annotated corpora); (C3) a working local multi-agent experimental prototype with Inventory, Evidence, and Governance agents plus a Human Review Gate; (C4) genuine LangChain-based LLM invocation with prompt templates, tool interfaces, structured-output parsing, and retries; (C5) genuine LangGraph orchestration with stateful nodes/edges, routing, and mandatory human-review control; (C6) an explicit separation of orchestration, LLM reasoning, deterministic safeguards, LangSmith trace capture, semantic evaluation, and human decision; (C7) a baseline-versus-experimental comparison against the unchanged SHADOWAI-RISK prototype; and (C8) an interpretive governance-control mapping. No claim is made that semantic preservation effectiveness has been proven in production; the computational matrix is complete (450/450), while human calibration on annotated corpora remains pending.

**Table 2.** Research questions.

| ID  | Research question                                                                                                                                    |
|-----|------------------------------------------------------------------------------------------------------------------------------------------------------|
| RQ1 | How can semantic meaning, constraints, and evidence commitments be represented across agent-to-agent handoffs?                                       |
| RQ2 | How can LangSmith traces be transformed into governance-relevant measures of preservation, drift, contradiction, and tool-use fidelity?              |
| RQ3 | Where do information loss, instruction drift, hallucination propagation, and failed handoffs threaten enterprise AI risk workflows?                  |
| RQ4 | How can SHADOWAI-RISK's evidence-linked governance pathway be extended by semantic tracing without conflating implemented and proposed capabilities? |
| RQ5 | How should semantic-tracing signals map interpretively to NIST, ISACA, EU, ENISA, OECD, and IEEE oversight expectations?                             |
| RQ6 | What evaluation protocol supports reproducible quantitative and qualitative assessment prior to, and after, empirical data collection?               |

## 4. Related Work

### 4.1 Multi-agent systems and LLM agents

Wooldridge provides foundational agent concepts [9]. Recent surveys document LLM-based agents and multi-agent organizations with natural-language channels [5–7,22]. Communication protocol design, role profiling, and evaluation remain open problems [8,21,23,24].

### 4.2 Hallucination and unreliable generation

Ji et al. survey hallucination in natural language generation [25–27]. In multi-agent settings the risk is compositional: an unsupported claim introduced by one agent can become an assumed premise for downstream agents [23,28,29]. This motivates an explicit Hallucination Propagation Rate.

### 4.3 Observability and tracing

LangSmith documentation describes traces as records of application steps for debugging, evaluation, and monitoring [12,30]. Xia et al. propose an AgentOps taxonomy of artifacts to trace across the agent lifecycle [13]. The present study uses these sources as infrastructure references and contributes semantic metrics and governance bindings that those sources do not define.

### 4.4 Governance, cybersecurity, and assurance

Relevant instruments include NIST AI RMF 1.0 and the Generative AI Profile [1,16]; NIST CSF 2.0 [31]; the EU AI Act [2]; OECD AI Principles [3]; ENISA AI cybersecurity and threat-landscape reports [4,14]; ISACA AI governance and audit guidance [18,19,44]; IEEE 7000-2021 [32]; ISO/IEC 42001 [33]; and ISO/IEC 23894 [50].

## 5. Semantic-Tracing Framework and Experimental Implementation

### 5.1 Semantic communication and handoff packets

Semantic communication, for this study, is the transfer of task-relevant meaning—goals, constraints, evidence commitments, uncertainty, and permitted actions—between agents such that a downstream agent can act without violating sender intent or organizational policy. Figure 3 presents the handoff packet schema.

Figure 2 presents the layered architecture and interfaces. Storage, policy checks, and human-review gates are first-class components rather than afterthoughts [34].

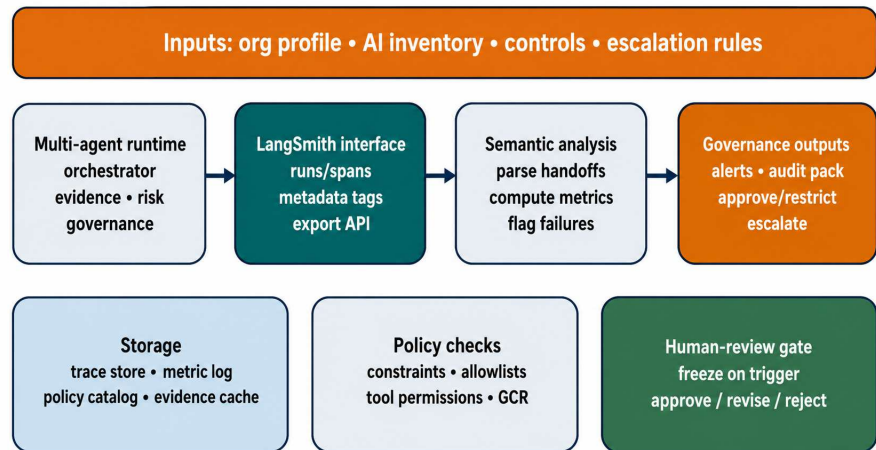


**Figure 2.** Semantic-tracing architecture with interfaces among multi-agent runtime, LangSmith, semantic analysis, storage, policy checks, and human-review gates.

The handoff packet is the unit of analysis (Figure 3) [11,23,35,51]. Required fields include goal identifier and statement; constraint set; evidence identifiers and sources; confidence; permitted tools; residual risk; escalation state; and policy identifier. Semantic failures include constraint deletion, unauthorized scope expansion, unsupported evidence claims, contradictory recommendations, incomplete tool results presented as complete, and silent suppression of escalation triggers.

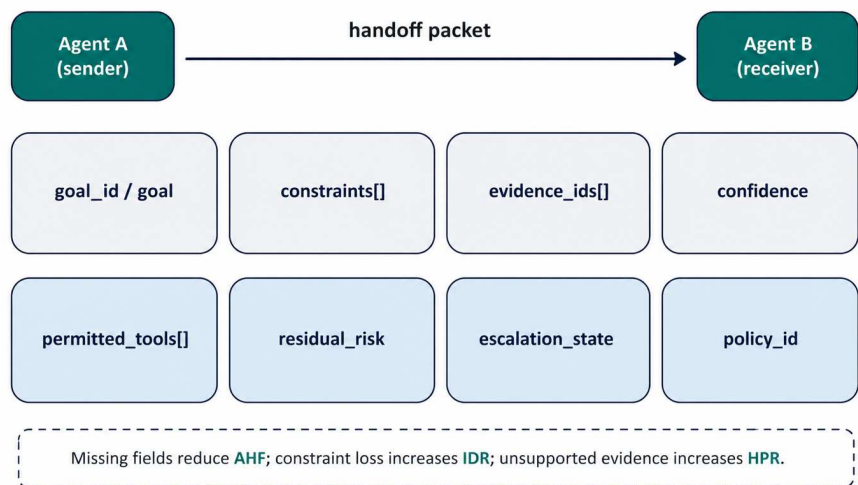


**Figure 3.** Agent-to-agent handoff packet schema with required fields used for fidelity and drift measurement.

5.2 LangSmith-based tracing pipeline

Figure 4 shows the pipeline: instrumentation, run/span capture, export, semantic parsing, metric calculation, and governance alert. Recommended metadata tags include agent\_role, hop\_index, policy\_id, tool\_name, evidence\_source, and human\_gate.

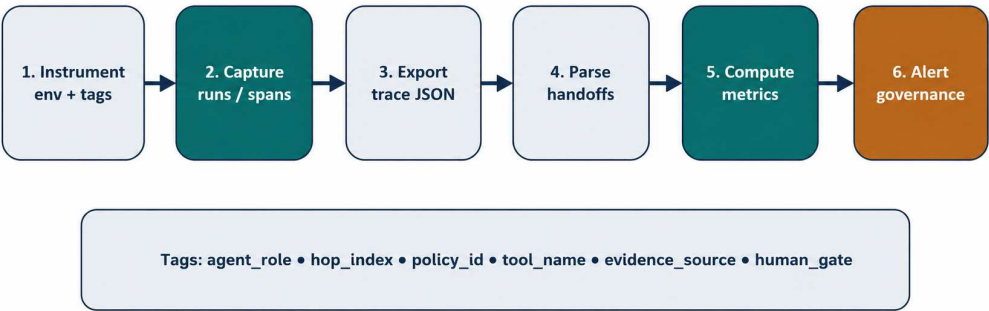


**Figure 4.** LangSmith pipeline from instrumentation through capture, export, semantic parsing, metric calculation, and governance alerting.

LangSmith provides trace capture for instrumented spans [12,36]; it does not automatically compute SPS or GCR. Those computations are study-specific services in the local experimental evaluation layer (implemented; calibration pending).

5.3 Failure taxonomy

Seven failure classes guide annotation and metric design [21]: (F1) constraint omission; (F2) scope expansion; (F3) evidence inflation; (F4) tool mismatch; (F5) handoff incompleteness; (F6) contradiction between agents; and (F7) escalation suppression. These classes map onto IDR, HPR, TUA, AHF, and HEP.

5.4 Experimental implementation

The semantic-tracing framework has been partially operationalized in a separate local experimental project (C:\Users\audre\OneDrive\Langsmith\SHADOWAI-RISK-LangSmith-Experimental). The original SHADOWAI-RISK production website remains unchanged; the experimental extension was developed locally and did not modify the Vercel deployment. No GitHub or Vercel actions were performed for this extension.

The experimental workflow implements four agent roles along Inventory → Evidence → Governance → Human Review under two executable conditions: (1) deterministic agents (agents/\*, workflow/run\_workflow.py) preserved as baseline; and (2) genuine LLM agents (llm\_agents/\*) invoked through LangChain and orchestrated by a LangGraph StateGraph (llm\_agents/graph\_langgraph.py). Handoffs are Pydantic-validated LLMHandoffPacket objects; evidence collection is exposed as LangChain tools (nvd\_cve\_lookup, evidence\_fixture\_loader); and LangSmith tracing is enabled through optional @traceable instrumentation when private credentials are present.

Functional testing confirms LangChain LLM calls, LangGraph orchestration, schema validation/repair, LangChain evidence tools, and optional private LangSmith spans. Implementation status (verified in code and smoke-tested): deterministic baseline complete; genuine LLM multi-agent implementation complete; LangChain integration implemented;

LangGraph orchestration implemented; LangSmith tracing implemented; full experimental matrix complete at 450/450 (complete:true; 0 terminal failures). Aggregate numerical and inferential results from the completed matrix are reported in Sections 8–12; human calibration of metric weights remains pending.

### 5.5 Framework roles and implementation evidence

Table 3 states each framework component’s actual role with code evidence. Only packages that are installed and imported in the local experimental project are claimed as integrated.

**Table 3.** Framework roles versus implementation evidence (packages genuinely used).

| Framework | Actual role in the system                                                                                                                   | Implementation evidence                                                                                                                                                                                                                                                                                                                           |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| LangChain | Real LLM invocation, prompt templates, tool interfaces, structured JSON output parsing, and retry/repair handling for each agent role.      | llm_agents/llm_client.py (ChatOpenAI, ChatPromptTemplate, JsonOutputParser, .with_retry, repair chain); llm_agents/prompts.py (langchain_role_prompt_template); llm_agents/tools_langchain.py (nvd_cve_lookup, evidence_fixture_loader); llm_agents/schema.py (LLMHandoffPacket); requirements.txt (langchain, langchain-core, langchain-openai). |
| LangGraph | Stateful multi-agent orchestration with workflow state, nodes, edges, conditional routing, and mandatory human-review control.              | llm_agents/graph_langgraph.py (StateGraph, nodes: inventory, evidence, governance, human_review; edges inventory→evidence→governance→human_review→END; add_conditional_edges for human-review routing); llm_agents/workflow_llm.py (run_langgraph_workflow); requirements.txt (langgraph).                                                        |
| LangSmith | Private tracing, span capture, run metadata/tags, and trace URL resolution. Does not compute semantic metrics or make governance decisions. | workflow/tracing.py (configure_langsmith, maybe_traceable, Client); llm_agents/workflow_llm.py (@traceable root SHADOWAI-RISK-LLM-Workflow + agent wrappers); private project semantic-tracing-shadowai.                                                                                                                                          |

### 5.6 System architecture (framework-integrated view)

Figure 5 summarizes the implemented execution pipeline: Scenario/Input → LangGraph Orchestrator (StateGraph) → LangChain-enabled Agents/Tools (ChatOpenAI, prompts, tools, JSON parse/retry) → LangSmith Observability → Semantic Evaluation Layer → Human Review Gate → Final Decision. LangSmith observes the workflow; it does not replace orchestration, evaluation, or human judgment.

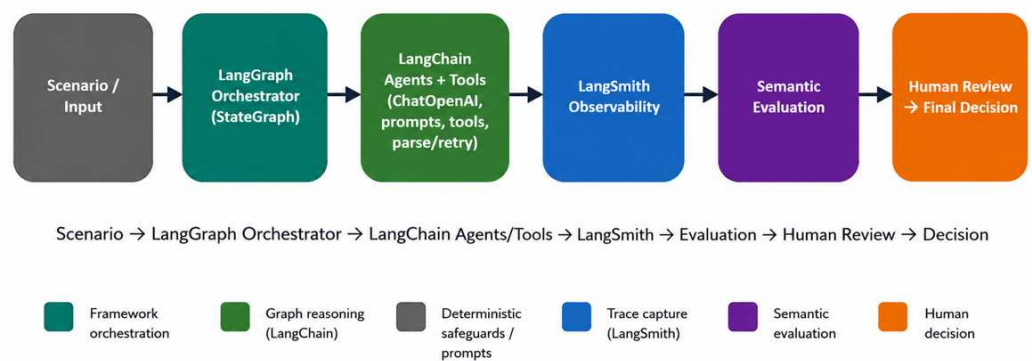


**Figure 5.** Implemented architecture pipeline: LangGraph orchestration, LangChain agents/tools, LangSmith trace capture, semantic evaluation, and human decision.

5.7 Distinguishing orchestration, reasoning, safeguards, tracing, evaluation, and human decision 175

Framework orchestration is LangGraph (11m\_agents/graph\_langgraph.py: nodes in- 176  
ventory/evidence/governance/human\_review and edges including mandatory routing to 177  
human review). LLM reasoning is LangChain ChatOpenAI invocation (11m\_agents/11m\_client.py 178  
with prompt templates from 11m\_agents/prompts.py. Deterministic safeguards include 179  
evidence collectors wrapped as LangChain tools, Pydantic schema coercion, and con- 180  
trolled experimental perturbations. Trace capture uses LangSmith (workflow/tracing.py, 181  
@traceable root/agent spans). Semantic evaluation is rule-based (agents/semantic\_detector.py) 182  
and is not attributed to LangSmith. Human decision is the Human Review Gate termi- 183  
nal node plus scenario controls; LangSmith does not decide approve/restrict/escalate 184  
outcomes. 185

6 Layered System Architecture 186

Figure 6 presents the overall system architecture and distinguishes the implemented 187  
SHADOWAI-RISK baseline, the locally implemented experimental extension, the external 188  
evidence sources, and the components that remain proposed. 189

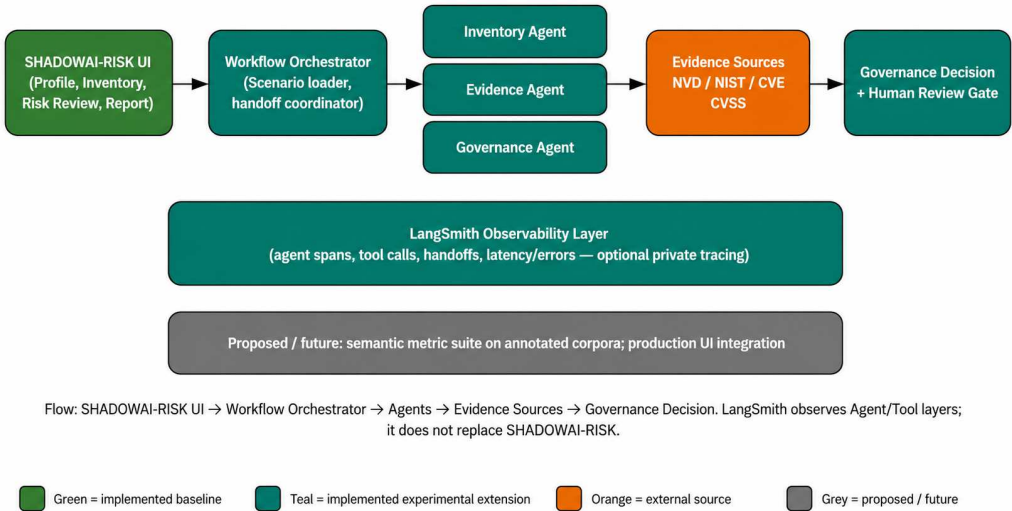


**Figure 6.** Overall system architecture showing SHADOWAI-RISK UI, experimental orchestra- 190  
tor/agents, external evidence sources, governance decision, and a LangSmith observability layer 191  
(implemented baseline in green; experimental extension in teal; external sources in orange; pro- 192  
posed/future in grey). 193

Figure 7 reorganizes the same architecture into six functional layers. Layer 1 contains 190  
the SHADOWAI-RISK user interface; Layer 2 provides workflow orchestration; Layer 191  
3 contains the Inventory, Evidence, and Governance agents; Layer 4 provides semantic 192  
tracing and observability; Layer 5 contains the cybersecurity evidence tools and sources; 193  
and Layer 6 stores workflow outputs, traces, handoff packets, governance decisions, and 194  
human-review status. 195

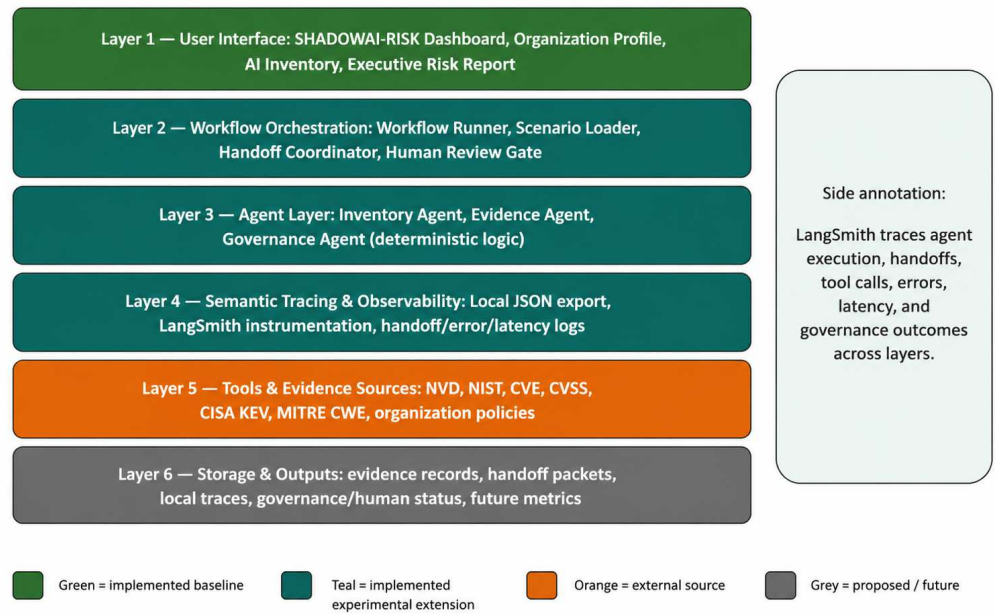


**Figure 7.** Layered system architecture (Layers 1–6) with LangSmith as a cross-cutting observability annotation.

LangSmith operates across these layers as an observability mechanism; it does not replace workflow orchestration, semantic evaluation, evidence retrieval, or human decision-making.

Figure 8 presents the implemented experimental workflow and shows how information moves from the organization profile through the Inventory, Evidence, and Governance agents to the Human Review Gate and final governance decision.

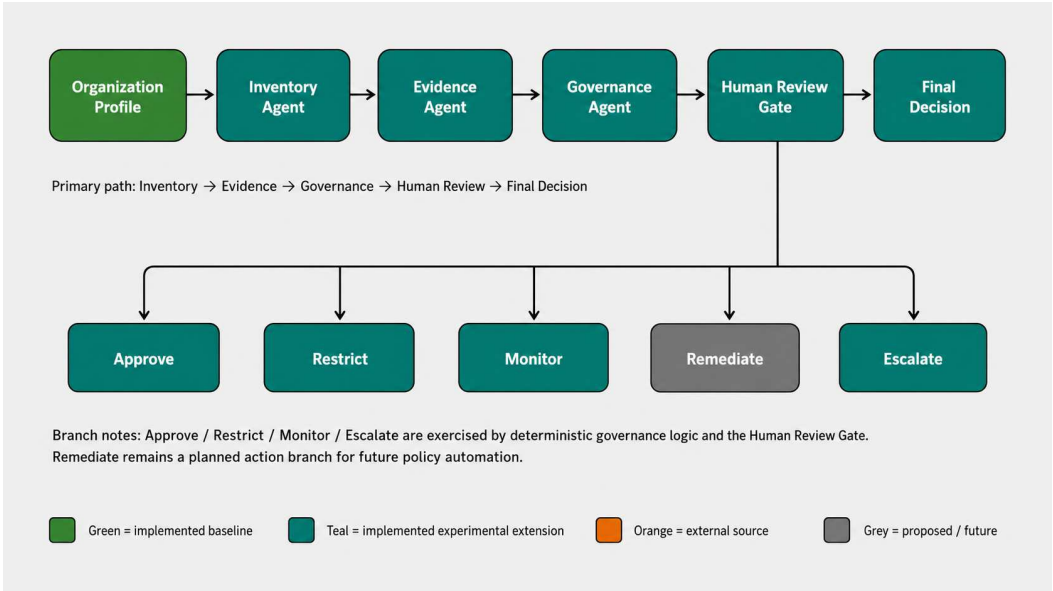


**Figure 8.** Experimental multi-agent workflow: Organization Profile → Inventory Agent → Evidence Agent → Governance Agent → Human Review Gate → Final Decision, with approve/restrict/monitor/escalate branches (remediate remains proposed).

The workflow makes the sequence of responsibility explicit: inventory information is collected first, evidence is retrieved and checked, governance recommendations are

generated, and cases requiring oversight are routed to human review. The remediate branch remains proposed and is not claimed as implemented.

Figure 9 specifies the structured handoff packet used to transfer goals, constraints, evidence references, confidence, residual risk, permitted actions, and escalation status between agents.

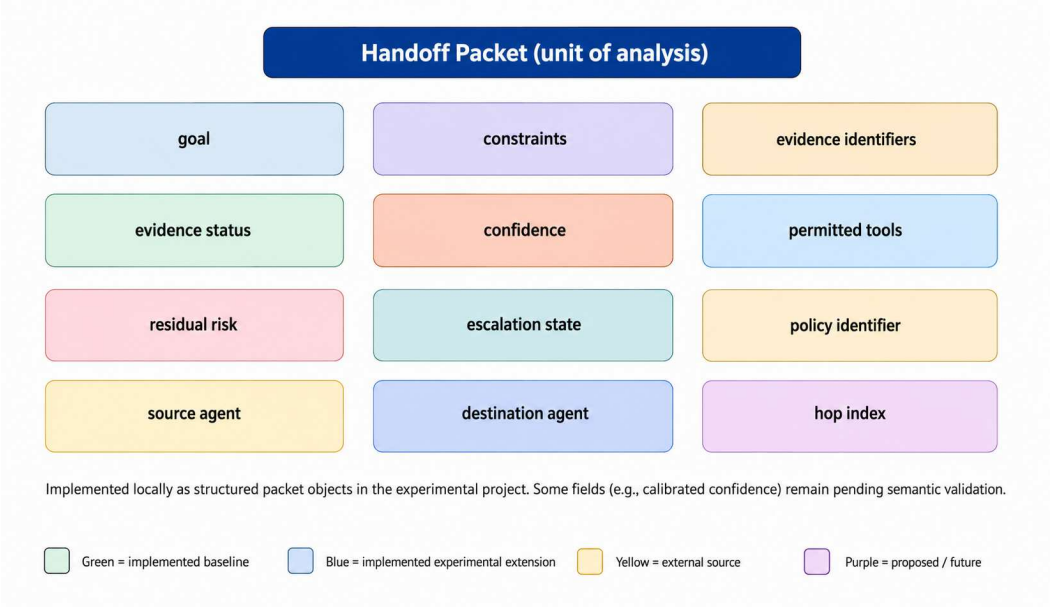


**Figure 9.** Agent handoff packet fields used by the experimental extension (implemented locally as structured objects).

These structured packets provide the observable unit used for comparison across workflow hops. Their fields allow the evaluation layer to identify missing constraints, unsupported evidence, scope expansion, and escalation-related changes.

Figure 10 illustrates the trace hierarchy produced by the implemented LangSmith instrumentation and its relationship to the local JSON trace export.

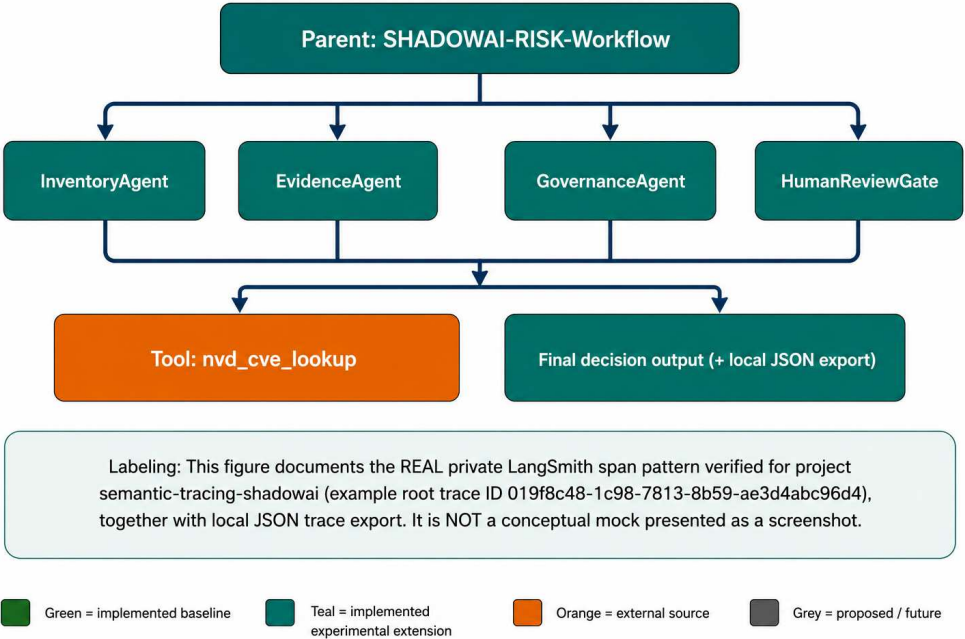


**Figure 10.** LangSmith trace structure corresponding to the verified private span pattern (project semantic-tracing-shadowai; example root trace ID 019f8c48-1c98-7813-8b59-ae3d4abc96d4). This figure documents a real private LangSmith pattern plus local JSON export; it is not a conceptual mock presented as a screenshot.

The trace structure separates the root workflow from agent-level and tool-level spans, allowing the execution path, metadata, latency, handoffs, and governance events to be inspected. LangSmith provides observability only; semantic scores and governance decisions are generated by separate system components.

Figure 11 compares the original deterministic SHADOWAI-RISK prototype with the separately implemented local multi-agent extension.

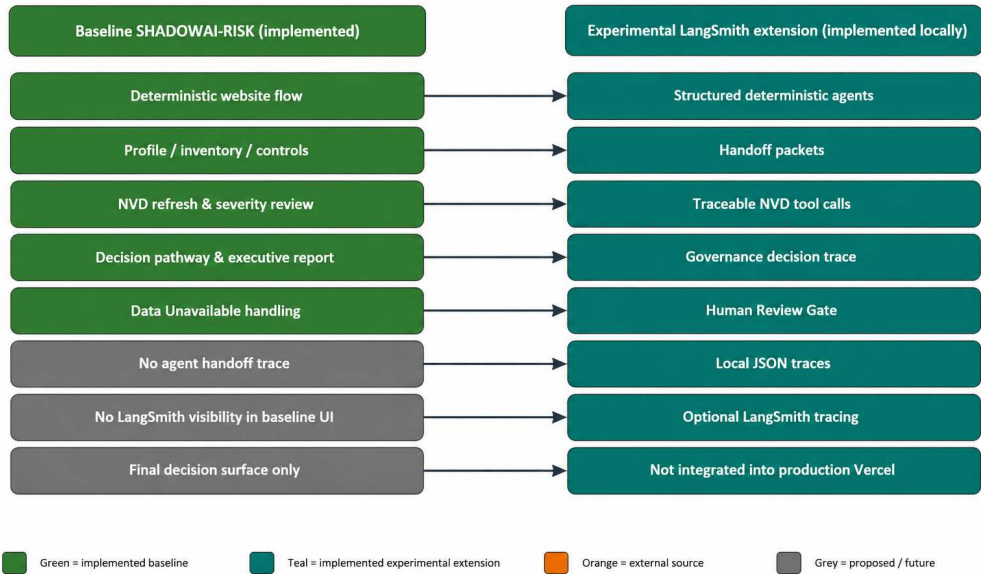


**Figure 11.** Baseline SHADOWAI-RISK versus the local experimental LangSmith extension (implemented locally; not integrated into production).

The comparison clarifies that the original website remains the deployed deterministic baseline, while the local extension adds agent orchestration, structured handoffs, trace capture, semantic evaluation, and human-review visibility. The experimental components were not integrated into the deployed Vercel interface.

## 7. Methodology

The methodology is mixed-methods and staged. Stage A develops the conceptual framework and metric formalization. Stage B analyzes the live SHADOWAI-RISK prototype. Stage C implements the local experimental extension (deterministic baseline preserved; LangChain LLM agents; LangGraph orchestration; LangSmith instrumentation; semantic evaluation). Stage D executes the controlled matrix with checkpointing and resume-safe skips of successful runs. Stage E finalized aggregate numerical reporting from the completed 450/450 matrix; human annotation for calibration and inter-rater agreement remains pending.

Integrity rules for this package are stated once here: (1) no invented participant statistics; (2) no invented metric cohort results before annotated controlled runs; (3) synthetic/conceptual traces are labeled as such, and real private LangSmith spans are identified by project/trace identifiers without presenting conceptual diagrams as screenshots; (4) citations are included only after DOI or official-URL verification; (5) baseline-implemented, experimentally implemented, and still-proposed components are separated in Table ?? and Figures 6–19. Semantic-metric datasets for Stage F and survey datasets for Stage G remain pending.

Related-work search prioritized peer-reviewed venues, major AI conferences, and official standards or regulatory sources (Appendix A). The complete literature matrix is provided as Supplementary File S1. The survey instrument is Supplementary File S2. Configuration templates, prompts, and metric stubs are Supplementary File S3. Editable figure sources are Supplementary File S4.

Figure 12 shows how qualitative and quantitative strands meet at validation.

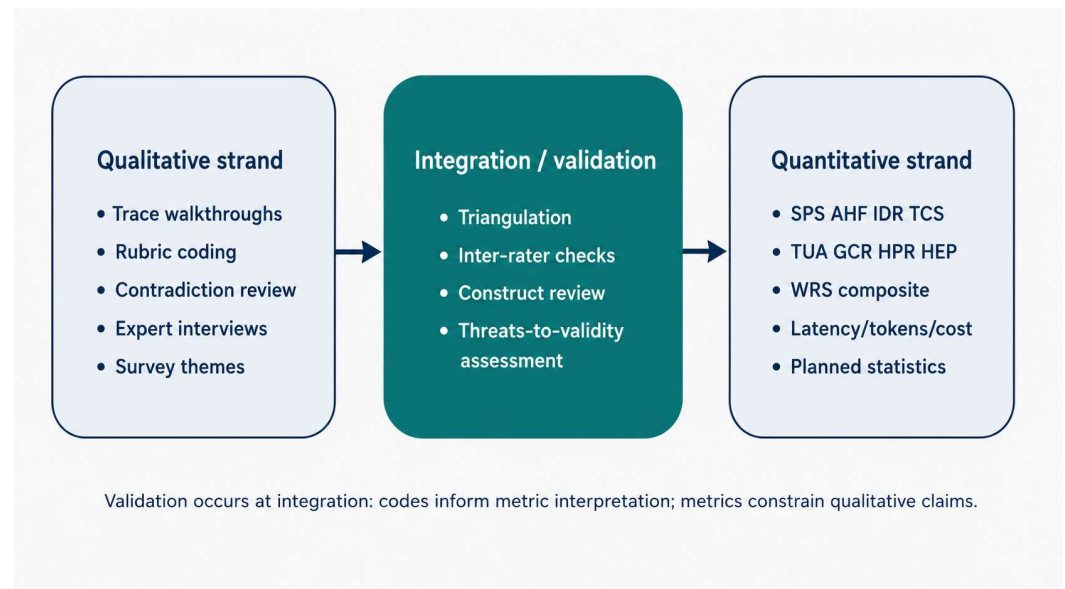
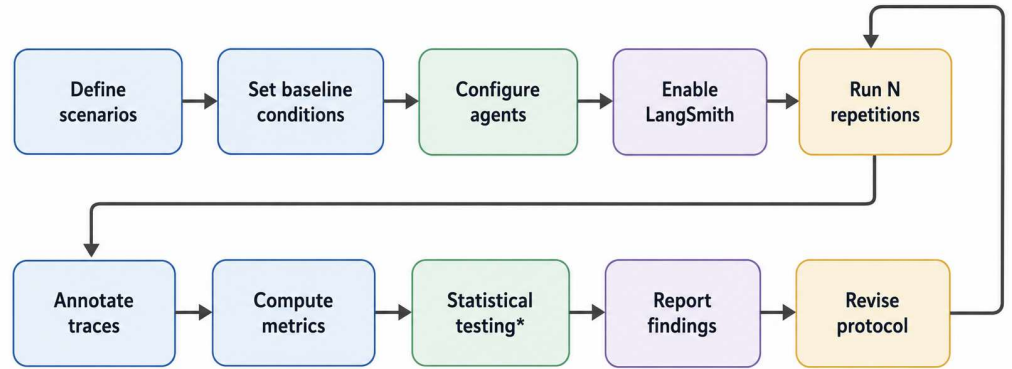


**Figure 12.** Mixed-methods evaluation design linking qualitative and quantitative strands at validation.

## 8. Experimental Design

Figure 13 summarizes the experimental loop. The expanded local design uses  $\geq 30$  controlled scenarios; three primary conditions (deterministic baseline; LLM without semantic evaluation; LLM with LangSmith + semantic evaluation); planned repetitions; temperature variability; and ablations. LLM conditions execute through LangGraph with LangChain agent/tool calls. The full experimental matrix completed at 450/450 (complete:true: deterministic 32, llm\_no\_eval 160, llm\_langsmith\_semantic\_eval 160, variability 50, ablations 48; 0 terminal failures).



Loop: revise protocol → rerun. \*Statistical testing is planned after data collection.

**Figure 13.** Experimental loop including repetitions, annotation, metric computation, planned statistical testing, and reporting.

Repetitions, baseline conditions, annotation, metric computation, and planned statistical testing are part of the loop. With the matrix complete (450/450), primary eval versus no-eval results ( $n=160$  each) are: TCS 0.624 vs. 0.912 (Holm-adjusted  $p < 0.001$ , Cliff  $\delta \approx -0.33$  medium); WRS\_equal and WRS\_expert significantly lower with eval (small effects); SPS/AHF/IDR/TUA/GCR/HPR/HEP not significant; latency  $\sim 126.2$  s vs.  $\sim 101.7$  s (significant, Cliff  $\delta \approx 0.67$  large); tokens not significant; local cost \$0; decision-family correctness 57/160 (0.356) vs. 59/160 (0.369), Fisher/ $\chi^2$  n.s. ( $p \approx 0.91$ ). Ablations ( $n=8$ , low power) associate removing constraints / evidence identifiers with large SPS drops (Cliff  $\delta \sim 0.5$ – $0.625$ ) and should be interpreted cautiously. Human calibration and inter-rater annotation remain pending.

Figure 14 visually summarizes the two primary reported outcomes. Trace completeness was lower in the evaluation-enabled condition, while decision-family correctness remained similar across conditions.

Figure 15 provides a conceptual nested-run schema for documentation only. It must not be read as an empirical LangSmith screenshot.

## 9. Metrics

Figure 16 shows how baseline intent is compared with each hop before metric calculation.

Table 4 presents the evaluation metric suite.

Let a workflow have ordered hops  $h = 0 \dots H$ , with reference intent  $I_0$  and observed intent  $I_h$ . Let  $C(I)$  be the constraint set, and let  $\text{EvidSupport}(I_h)$  penalize unsupported claims.

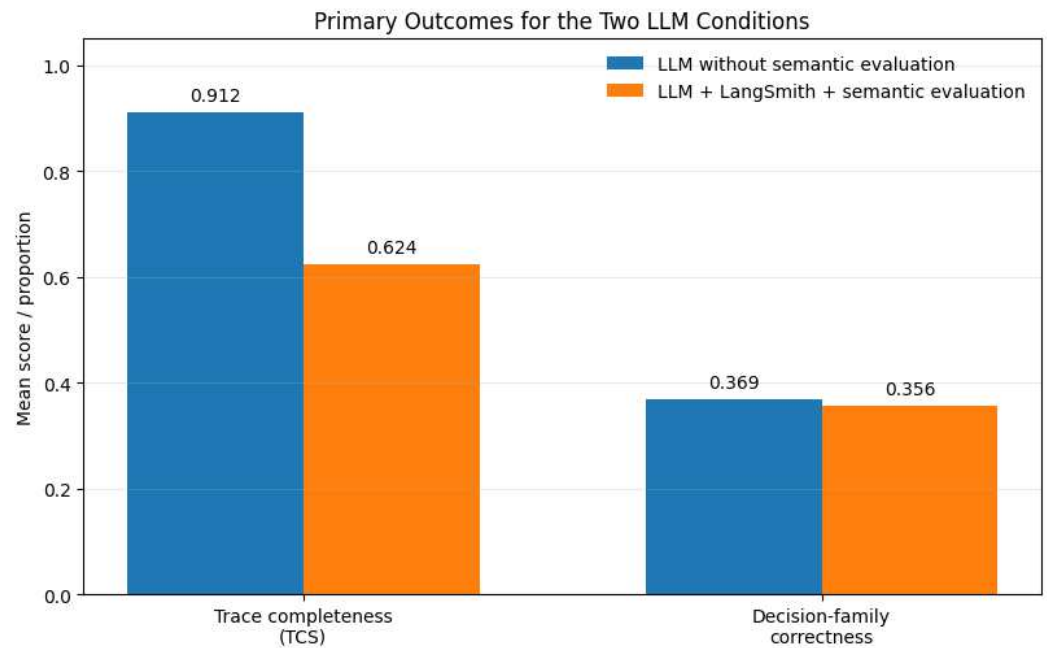


**Figure 14.** Comparison of the primary outcomes for the two LLM conditions. The LLM condition without semantic evaluation showed higher trace completeness than the LLM condition with LangSmith and semantic evaluation, whereas decision-family correctness was similar between conditions. Each condition included 160 runs.

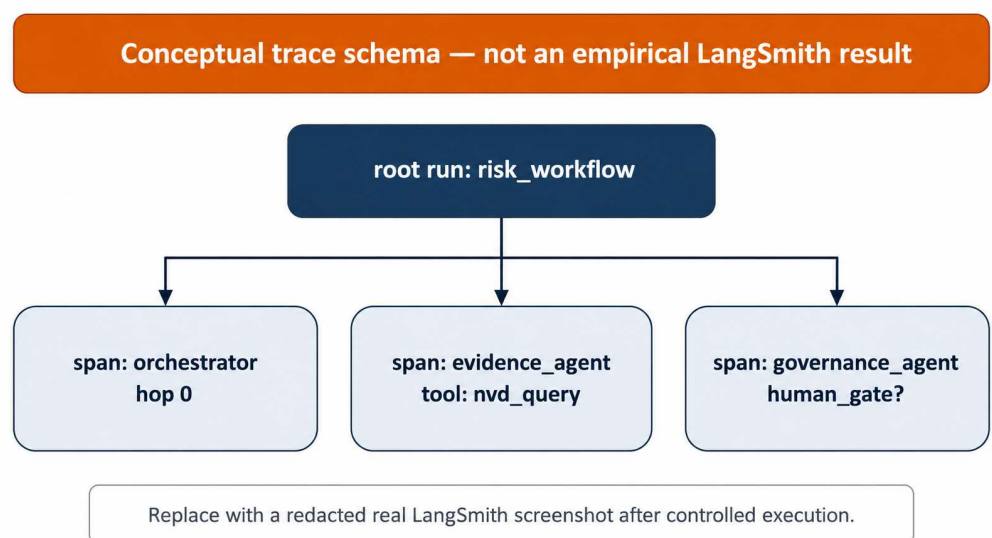


**Figure 15.** Conceptual nested-run schema for documentation. Not an empirical LangSmith result.

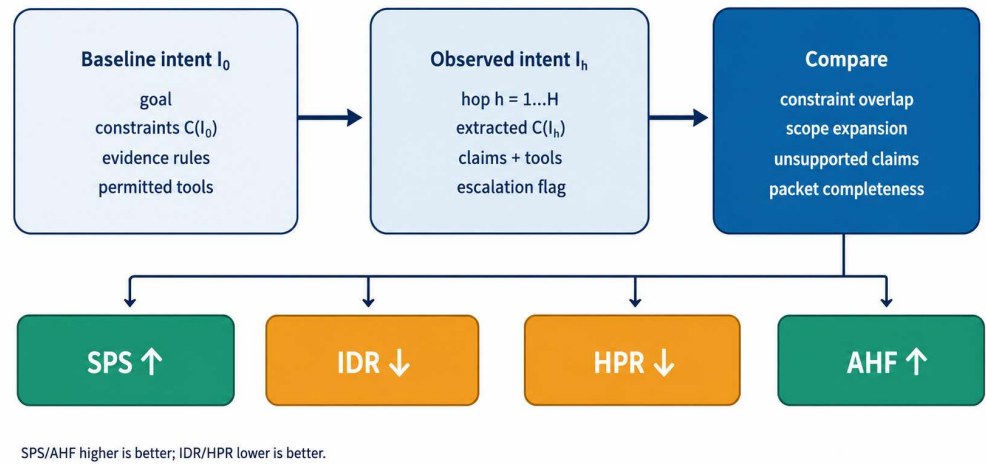


**Figure 16.** Comparison of baseline intent with subsequent hops and derivation of SPS, IDR, HPR, and AHF.

**Table 4.** Evaluation metric suite.

| Metric                         | Abbrev. | Range | Interpretation                                  |
|--------------------------------|---------|-------|-------------------------------------------------|
| Semantic Preservation Score    | SPS     | [0,1] | Higher = better meaning/constraint preservation |
| Agent Handoff Fidelity         | AHF     | [0,1] | Higher = more complete valid handoff packets    |
| Instruction Drift Rate         | IDR     | [0,1] | Higher = more drift (worse)                     |
| Trace Completeness Score       | TCS     | [0,1] | Higher = more required spans/fields present     |
| Tool-Use Accuracy              | TUA     | [0,1] | Higher = correct tool selection/execution       |
| Governance Compliance Rate     | GCR     | [0,1] | Higher = fewer policy violations                |
| Hallucination Propagation Rate | HPR     | [0,1] | Higher = more propagated unsupported claims     |
| Human Escalation Precision     | HEP     | [0,1] | Higher = escalations are warranted              |
| Workflow Reliability Score     | WRS     | [0,1] | Composite reliability indicator                 |

$$\text{SPS}_h = \alpha \cdot \text{Sim}(I_0, I_h) + \beta \cdot \frac{|C(I_0) \cap C(I_h)|}{|C(I_0)|} + \gamma \cdot \text{EvidSupport}(I_h), \quad \alpha + \beta + \gamma = 1 \quad (1)$$

Equation (1) defines the Semantic Preservation Score (SPS) for handoff  $h$ . It measures how well the observed downstream intent preserves the original intent. The metric combines semantic similarity, constraint-set overlap, and evidence support. The weights  $\alpha$ ,  $\beta$ , and  $\gamma$  determine the contribution of the three components and sum to one. A higher SPS value indicates stronger preservation of the original intent across the handoff. The workflow-level SPS is the mean of  $\text{SPS}_h$  over all evaluated handoffs  $h \geq 1$ .

$$\text{AHF} = \frac{1}{H} \sum_h \text{Completeness}(\text{packet}_h) \cdot \text{Validity}(\text{packet}_h) \quad (2)$$

Equation (2) defines Agent Handoff Fidelity (AHF). AHF measures the completeness and validity of handoff packets. A higher AHF value is preferable and indicates that packets are both more complete and more valid across hops.

$$\text{IDR} = \text{clip}_{[0,1]} \left( \frac{1}{H} \sum_h \left[ 1 - \frac{|C(I_0) \cap C(I_h)|}{|C(I_0)|} + \text{ScopeExpansion}(I_h) \right] \right) \quad (3)$$

Equation (3) defines the Instruction Drift Rate (IDR). IDR measures instruction or constraint drift and scope expansion. A lower IDR value is preferable and indicates less drift relative to the reference intent.

$$\text{TCS} = \frac{|\text{ObservedRequiredSpans} \cap \text{RequiredSpans}|}{|\text{RequiredSpans}|} \quad (4)$$

Equation (4) defines the Trace Completeness Score (TCS). TCS measures the completeness of required trace spans. A higher TCS value is preferable and indicates that a larger fraction of required spans is observed.

$$\text{TUA} = \frac{\# \text{ correct tool decisions}}{\# \text{ tool decisions}} \quad (5)$$

Equation (5) defines Tool-Use Accuracy (TUA). TUA measures the proportion of correct tool-use decisions. A higher TUA value is preferable and indicates more accurate tool selection and execution.

$$\text{GCR} = 1 - \frac{\# \text{ policy violations}}{\# \text{ policy checks}} \quad (6)$$

Equation (6) defines the Governance Compliance Rate (GCR). GCR measures policy compliance. A higher GCR value is preferable and indicates fewer policy violations relative to the number of policy checks.

$$\text{HPR} = \frac{\# \text{ unsupported claims reused downstream}}{\# \text{ unsupported claims introduced}} \quad (7)$$

Equation (7) defines the Hallucination Propagation Rate (HPR). HPR measures the downstream reuse of unsupported claims. A lower HPR value is preferable and indicates less propagation of unsupported claims across agent hops.

$$\text{HEP} = \frac{\# \text{ warranted escalations}}{\# \text{ escalations raised}} \quad (8)$$

Equation (8) defines Human Escalation Precision (HEP). HEP measures the proportion of warranted escalations among all escalations raised. A higher HEP value is preferable and

indicates that escalations more consistently correspond to conditions that warrant human review.

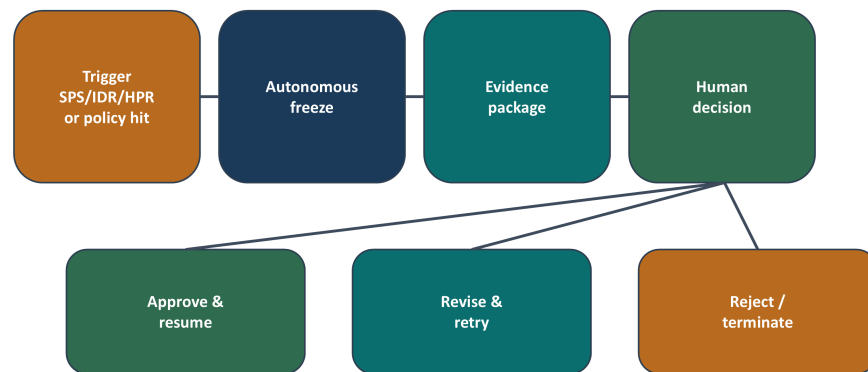
$$\text{WRS} = \text{normalize}(w_1 \cdot \text{SPS} + w_2 \cdot \text{AHF} + w_3 \cdot \text{TCS} + w_4 \cdot \text{TUA} + w_5 \cdot \text{GCR} - w_6 \cdot \text{IDR} - w_7 \cdot \text{HPR}) \quad (9)$$

Equation (9) defines the Workflow Reliability Score (WRS). WRS combines the normalized metrics into one weighted workflow-level score. A higher WRS value is preferable and indicates stronger overall workflow reliability under the chosen weighting scheme.

Taken together, SPS and AHF evaluate semantic preservation and handoff quality; TCS and TUA evaluate trace and tool-use correctness; GCR evaluates policy compliance; IDR and HPR capture undesirable drift and unsupported-claim propagation; HEP evaluates escalation quality; and WRS combines the normalized metrics into an overall workflow score.

Weight values require calibration on future annotated corpora [24,42]. Latency, token usage, and cost per workflow are logged from LangSmith metadata as operational complements, not semantic fidelity measures. Additional formal notes appear in Appendix C.

Figure 17 shows the human-escalation process tied to metric and policy triggers [34,43].



Example triggers (to be calibrated): low SPS, high IDR/HPR, missing evidence, tool-permission breach.

**Figure 17.** Human-in-the-loop escalation process with trigger, freeze, evidence package, human decision, and resume or terminate paths.

## 10. Results

### 10.1 Case-study observational results

Inspection of the live SHADOWAI-RISK prototype confirms an academic-research-prototype framing and an executive workflow organized around profile setup, intelligence refresh, risk-finding review, and report generation. Live-review priority indicators depend on successful refresh against the security database. When refresh has not occurred or source values are missing, fields are shown as Data Unavailable or Not refreshed. The decision pathway sequences Known Security Issue → Affected Evidence → Risk Level → What To Do.

## 10.2 Framework and instrumentation products

Stage A/B products include the metric specification, trace-tagging scheme, evaluation rubric, and the three-category implemented/experimental/proposed boundary (Figures 6–11; Table ??). Stage C/D/E products include the local experimental project, functional tests, local JSON traces, and optional private LangSmith spans. These products support process-level inspection; they do not by themselves constitute proven semantic-preservation effectiveness.

### 10.2.1 Implementation details (code-backed)

LLM invocation uses LangChain ChatOpenAI against a local Ollama OpenAI-compatible endpoint by default (`llm_agents/llm_client.py`), with `ChatPromptTemplate` and `JsonOutputParser`, plus `.with_retry` and a repair chain. Evidence tooling is exposed as LangChain @tool functions in `llm_agents/tools_langchain.py`. Orchestration is a compiled LangGraph StateGraph in `llm_agents/graph_langgraph.py` with nodes `inventory/evidence/governance/human_review`. LangSmith root/agent spans are created via `langsmith.traceable` when enabled. Semantic evaluation remains rule-based (`agents/semantic_detector.py`) and is not attributed to LangSmith.

### 10.3 Genuine LLM-based multi-agent condition (local experimental extension)

The completed experimental matrix comprised 450 workflow executions, including 160 runs without semantic evaluation and 160 runs with LangSmith tracing and semantic evaluation. The primary quantitative comparison is summarized below.

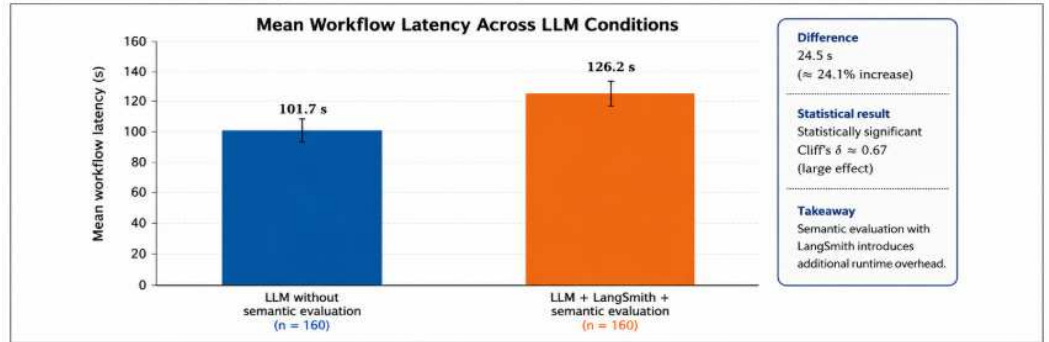


Table 6. Summary of the primary comparison between the two LLM conditions.

| Outcome                                     | LLM without evaluation (n = 160) | LLM + LangSmith + semantic evaluation (n = 160) | Differences              | Statistical result                                                  | Takeaway                                                             |
|---------------------------------------------|----------------------------------|-------------------------------------------------|--------------------------|---------------------------------------------------------------------|----------------------------------------------------------------------|
| Trace Completeness Score (TCS)              | 0.912                            | 0.624                                           | -0.288                   | Holm-adjusted $p < 0.001$ ; Cliff's $\delta \approx -0.33$ (medium) | Lower TCS in evaluation condition.                                   |
| Workflow Reliability Score (equal weights)  | Not separately reported          | Not separately reported                         | Lower                    | Significant after Holm correction                                   | Lower WRS (equal weights) with evaluation.                           |
| Workflow Reliability Score (expert weights) | Not separately reported          | Not separately reported                         | Lower                    | Significant after Holm correction                                   | Lower WRS (expert weights) with evaluation.                          |
| SPS, AHF, IDR, TUA, GCR, HPR, and HEP       | Computationally evaluated        | Computationally evaluated                       | No meaningful difference | Not significant                                                     | No significant differences in these semantic and governance metrics. |
| Decision-family correctness                 | 59/160 (0.369)                   | 57/160 (0.356)                                  | -0.013                   | Fisher / $\chi^2$ $p \approx 0.91$                                  | Similar decision-family correctness.                                 |
| Mean workflow latency (s)                   | 101.7 s                          | 126.2 s                                         | +24.5 s (+24.1%)         | Significant; Cliff's $\delta \approx 0.67$ (large)                  | Semantic evaluation increased workflow latency.                      |
| Token usage                                 | Recorded                         | Recorded                                        | No meaningful difference | Not significant                                                     | No significant difference in token usage.                            |
| Local execution cost (USD)                  | \$0                              | \$0                                             | —                        | Not applicable                                                      | Both conditions ran locally at zero cost.                            |

**Figure 18.** Comparison of the two primary LLM conditions. The semantic-evaluation condition increased workflow latency while maintaining similar decision-family correctness.

A genuine LLM-based multi-agent condition was implemented with LangChain and LangGraph and compared with the deterministic baseline under controlled scenarios. Each of the four roles issues an actual language-model API call through LangChain ChatOpenAI and returns a Pydantic-validated handoff packet; hops are scheduled by LangGraph. The deterministic baseline remains runnable and unmodified for comparison. The completed 450/450 matrix supports primary eval versus no-eval contrasts ( $n=160$  each) under this design, with aggregate inferential results reported elsewhere in this manuscript

**Table 5.** Summary of the primary comparison between the LLM multi-agent condition without semantic evaluation and the LLM condition with LangSmith tracing and semantic evaluation. Each primary condition included 160 runs.

| Outcome                                     | LLM without evaluation | LLM with evaluation | Statistical result                  | Effect size                             | Interpretation                                                                                    |
|---------------------------------------------|------------------------|---------------------|-------------------------------------|-----------------------------------------|---------------------------------------------------------------------------------------------------|
| Trace Completeness Score (TCS)              | 0.912                  | 0.624               | Holm-adjusted $p < 0.001$           | Cliff's $\delta \approx -0.33$ (medium) | Trace completeness was significantly lower in the evaluation-enabled condition.                   |
| Workflow Reliability Score (equal weights)  | NR                     | NR                  | Significant after Holm correction   | Small                                   | WRS under equal weighting was significantly lower with semantic evaluation.                       |
| Workflow Reliability Score (expert weights) | NR                     | NR                  | Significant after Holm correction   | Small                                   | WRS under expert-defined weighting was significantly lower with semantic evaluation.              |
| SPS, AHF, IDR, TUA, GCR, HPR, and HEP       | Computed               | Computed            | Not significant                     | No meaningful effect established        | No statistically significant differences were observed for these semantic and governance metrics. |
| Decision-family correctness                 | 59/160 (0.369)         | 57/160 (0.356)      | Fisher/ $\chi^2$ , $p \approx 0.91$ | Negligible                              | Decision-family correctness was similar between the two conditions.                               |
| Mean workflow latency                       | $\sim 101.7$ s         | $\sim 126.2$ s      | Significant                         | Cliff's $\delta \approx 0.67$ (large)   | Semantic evaluation increased workflow latency.                                                   |
| Token usage                                 | Recorded               | Recorded            | Not significant                     | No meaningful effect established        | Token usage did not differ significantly between conditions.                                      |
| Local execution cost                        | \$0                    | \$0                 | Not applicable                      | Not applicable                          | Both conditions used local model execution and incurred no API cost.                              |

NR = Not separately reported.

Figure 18 provides a visual summary of the principal performance difference between the two experimental conditions, whereas Table 5 reports the complete quantitative and statistical comparison across all evaluated metrics.

Study design uses three conditions on the same expanded scenario set ( $\geq 30$  scenarios): (A) deterministic baseline; (B) LLM multi-agent without semantic evaluation; (C) LLM multi-agent with private LangSmith tracing and a semantic evaluation layer. This separation is required so improvements from evaluation logic are not attributed to LangSmith observability alone.

Expanded scenarios include clean evidence, low-risk approval, high-risk restriction, missing/conflicting evidence, unsupported recommendation, instruction drift, constraint omission, scope expansion, evidence inflation, tool mismatch/failure, Shadow AI, mandatory escalation, and escalation suppression, with mild/moderate/severe drift severities and multiple hop locations. LLM scenarios are configured for repeated runs (target  $\geq 5$ ) with temperature variability experiments. Deterministic runs are executed once per scenario because the baseline is fully reproducible.

Metric methodology was strengthened beyond one-off prototype scoring: formal numerator/denominator definitions, missing-data rules, aggregation, confidence intervals, and WRS sensitivity across equal, expert-defined, and data-driven weights are specified in `Metric_Definitions_and_Calibration.md`. Human annotation guidelines were prepared; inter-rater agreement is pending and is not fabricated. Therefore, the metrics are treated as computationally applied under formal definitions rather than as fully calibrated or externally validated measures.

Cautious interpretation is required. LangSmith records the workflow, whereas the evaluation layer compares handoff states, evidence status, and governance outcomes. This manuscript does not claim proven semantic preservation, production readiness, or large-scale statistical significance on the basis of the local controlled experiment alone.

Figure 19 summarizes the overall three-condition study design. It separates the deterministic baseline, the LLM multi-agent condition without semantic evaluation, and the LLM multi-agent condition with LangSmith tracing and semantic evaluation.

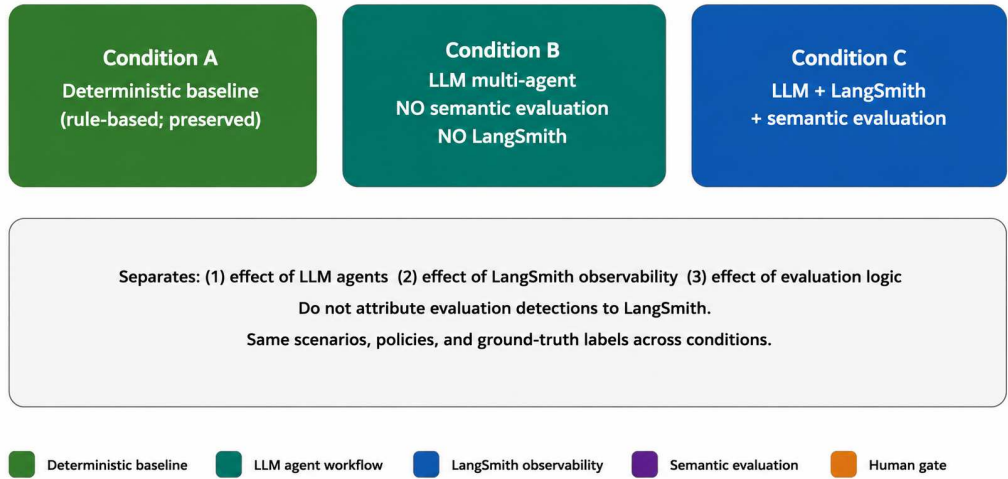


Figure 19. Three-condition study design separating agents, LangSmith, and evaluation.

This separation prevents the effects of semantic evaluation from being incorrectly attributed to LangSmith observability alone and clarifies the role of each experimental component.

Figure 20 presents the implemented LLM-based multi-agent workflow and the schema-validated exchange of information across the Inventory, Evidence, Governance, and Human Review stages.

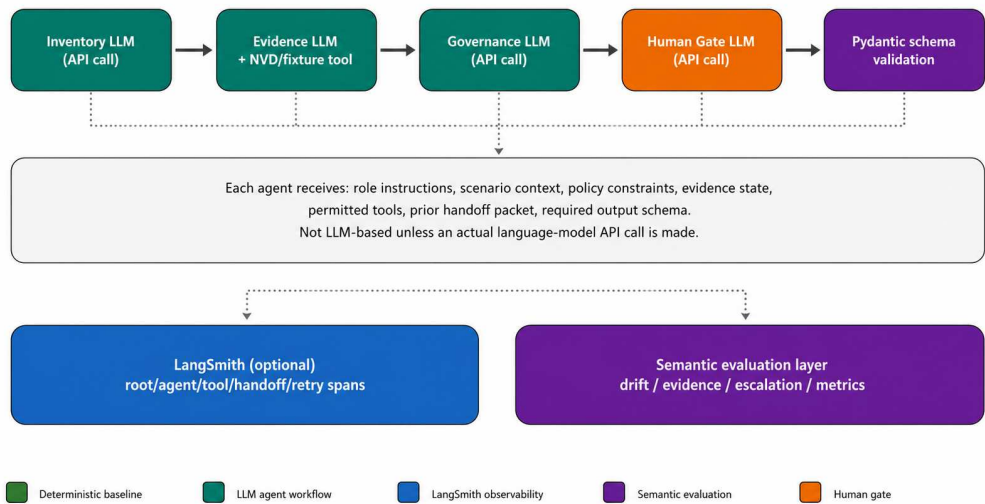


Figure 20. Genuine LLM multi-agent workflow with schema validation.

The workflow shows how each agent produces a structured handoff packet before the next stage executes, allowing the communication path to be inspected and validated.

Figure 21 summarizes the metric hierarchy used in the study and distinguishes computationally applied metrics from measures that remain pending human calibration.

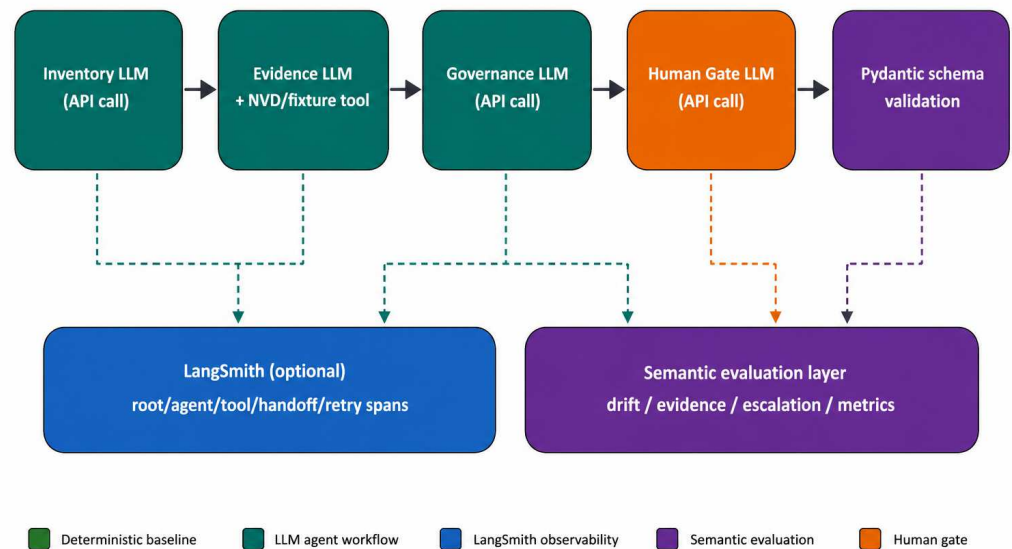


**Figure 21.** Metric stack distinguishing computational scores from pending human calibration.

This distinction prevents prototype computational measures from being interpreted as externally validated governance indicators.

A small controlled laboratory comparison was executed locally between (i) the deterministic SHADOWAI-RISK decision pathway used by the current application logic and (ii) the LangSmith-instrumented multi-agent workflow (Inventory Agent → Evidence Agent → Governance Agent → Human Review Gate). The production Vercel deployment and GitHub repository were not modified.

**Baseline definition.** The baseline is a local deterministic port of SHADOWAI-RISK application logic: organization profile and AI inventory, NVD/NIST evidence attachment, severity-based recommended actions, and Data Unavailable handling. The baseline does not implement multi-agent constraint handoffs, LangSmith tracing, or semantic-drift detection.

**Experimental-condition definition.** The experimental condition uses structured hand-off packets, private LangSmith tracing, and a rule-based evaluation layer. LangSmith records the workflow; it does not make governance decisions. The evaluation layer compares handoff states, evidence status, and governance outcomes to identify drift, unsupported recommendations, and escalation conditions.

**Scenario set ( $n = 5$ ).** The controlled scenarios included clean valid evidence; instruction drift (omission of the missing-evidence constraint at hop 1); missing evidence (forced Data Unavailable); mandatory human escalation (unapproved Shadow AI); and conflicting recommendations (contradictory channels). Intentionally introduced errors were labeled as controlled experimental perturbations.

Table 6 reports the expected and observed decisions, escalation behavior, drift detection, evidence status, runtime, and overall result for each scenario and condition.

**Table 6.** Per-scenario comparison of the deterministic baseline and multi-agent conditions.

| (a) Decision outcomes by scenario                    |                            |               |                |                     |                      |        |
|------------------------------------------------------|----------------------------|---------------|----------------|---------------------|----------------------|--------|
| Scenario                                             | Condition                  | Expected      | Actual         | Escalation expected | Escalation performed |        |
| Clean valid-evidence scenario                        | Deterministic baseline     | Monitor       | Monitor        | False               | False                |        |
| Clean valid-evidence scenario                        | Multi-agent with LangSmith | Proceed       | Proceed        | False               | False                |        |
| Intentionally introduced instruction-drift scenario  | Deterministic baseline     | Monitor       | Monitor        | False               | False                |        |
| Intentionally introduced instruction-drift scenario  | Multi-agent with LangSmith | Hold          | Hold           | True                | True                 |        |
| Missing-evidence scenario                            | Deterministic baseline     | Monitor       | Monitor        | False               | False                |        |
| Missing-evidence scenario                            | Multi-agent with LangSmith | Hold          | Hold           | True                | True                 |        |
| Mandatory human-escalation scenario                  | Deterministic baseline     | Escalate      | Escalate       | True                | True                 |        |
| Mandatory human-escalation scenario                  | Multi-agent with LangSmith | Reject        | Reject         | True                | True                 |        |
| Conflicting-recommendation scenario                  | Deterministic baseline     | Escalate      | Escalate       | True                | True                 |        |
| Conflicting-recommendation scenario                  | Multi-agent with LangSmith | Hold          | Hold           | True                | True                 |        |
|                                                      |                            |               |                |                     |                      |        |
| (b) Detection, evidence, runtime, and overall result |                            |               |                |                     |                      |        |
| Scenario                                             | Condition                  | Drift present | Drift detected | Evidence complete   | Runtime (s)          | Result |
| Clean valid-evidence scenario                        | Deterministic baseline     | False         | False          | True                | 0.0001               | PASS   |
| Clean valid-evidence scenario                        | Multi-agent with LangSmith | False         | False          | True                | 5.9330               | PASS   |
| Intentionally introduced instruction-drift scenario  | Deterministic baseline     | True          | False          | False               | 0.0001               | PASS   |
| Intentionally introduced instruction-drift scenario  | Multi-agent with LangSmith | True          | True           | False               | 6.1636               | PASS   |
| Missing-evidence scenario                            | Deterministic baseline     | False         | False          | False               | 0.0000               | PASS   |
| Missing-evidence scenario                            | Multi-agent with LangSmith | False         | False          | False               | 6.4757               | PASS   |
| Mandatory human-escalation scenario                  | Deterministic baseline     | False         | False          | False               | 0.0000               | PASS   |
| Mandatory human-escalation scenario                  | Multi-agent with LangSmith | False         | False          | False               | 5.1679               | PASS   |
| Conflicting-recommendation scenario                  | Deterministic baseline     | False         | False          | False               | 0.0001               | PASS   |
| Conflicting-recommendation scenario                  | Multi-agent with LangSmith | False         | False          | False               | 5.5164               | PASS   |

The prototype scenario-level metric rules were defined as follows: SPS equals 1 when no drift is present and otherwise equals  $\max(0, 1 - \text{lost}/|C_0|)$ ; AHF is the mean completeness of required handoff fields; IDR equals  $\text{lost}/|C_0|$  when drift is present and 0 otherwise; TCS approximates packet and trace completeness; TUA penalizes invented CVE identifiers; GCR equals 1 when unsupported claims are absent and 0 otherwise; HPR equals 1 when unsupported claims propagate and 0 otherwise; and HEP equals 1 when required escalations are handled by a human or when a non-escalated case is correctly left un-escalated. These prototype computations are not calibrated or externally validated.

Table 7 summarizes the practical measures obtained from the controlled prototype comparison.

**Table 7.** Summary of practical measures from the controlled prototype experiment. The measures are not externally calibrated.

| Measure                         | Deterministic baseline | Multi-agent with LangSmith | Interpretation                                                                                                    |
|---------------------------------|------------------------|----------------------------|-------------------------------------------------------------------------------------------------------------------|
| Decision accuracy               | 1.0                    | 1.0                        | Fraction of decisions matching the condition-specific expected outcome.                                           |
| Drift-detection recall          | 0.0                    | 1.0                        | Recall calculated for scenarios in which ground-truth drift was present. The baseline contains no drift detector. |
| Missing-evidence detection rate | 1.0                    | 1.0                        | Fraction of missing-evidence scenarios correctly marked as incomplete or unavailable.                             |
| Escalation precision            | 1.0                    | 1.0                        | Fraction of performed escalations that were expected under the scenario protocol.                                 |
| Escalation recall               | 1.0                    | 1.0                        | Fraction of expected escalations that were performed.                                                             |
| Average runtime (s)             | 0.0001                 | 5.8513                     | The multi-agent condition includes model execution, tool use, orchestration, and tracing overhead.                |
| Trace completeness              | 0.0                    | 1.0                        | The deterministic baseline does not generate LangSmith traces.                                                    |

The comparison showed high decision accuracy for both conditions relative to their condition-specific expected mappings. The multi-agent condition uniquely detected the controlled instruction-drift perturbation and recorded formal Human Review Gate outcomes. The deterministic baseline correctly handled Data Unavailable without inventing evidence and escalated Shadow AI or high-severity cases through deterministic rules, but it could not detect hop-level constraint drift because it does not contain constraint-carrying agent handoffs. The multi-agent runs were slower and more complex. No claim is made of production validation, proven semantic preservation, or large-scale statistical significance.

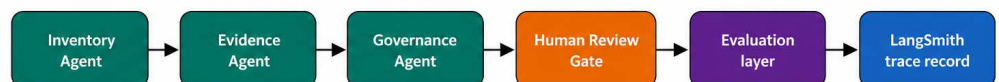
Figure 22 compares the deterministic baseline and the experimental multi-agent workflow at the process level.

#### Baseline (deterministic SHADOWAI-RISK)



No handoff packets · No drift detector · No LangSmith spans · No formal freeze gate

#### Experimental (multi-agent + LangSmith)



LangSmith records spans; evaluation logic compares handoffs/evidence/outcomes. LangSmith does not make decisions.

Fair comparison note: both conditions use equivalent inputs, evidence fixtures/force modes, and policies.

Drift perturbations apply only where multi-agent constraint handoffs exist.

Baseline missing-evidence behavior = Data Unavailable + monitor (not silent invent).

■ Deterministic app logic   
 ■ Agent workflow   
 ■ LangSmith observability   
 ■ Evaluation logic   
 ■ Human decision

**Figure 22.** Deterministic baseline versus multi-agent experimental workflow.

The comparison highlights that the multi-agent extension introduces explicit agent orchestration, structured handoffs, trace capture, semantic evaluation, and human-review visibility while preserving the governance objectives of the original deterministic pathway.

Figure 23 presents the controlled scenario design and the intentionally introduced perturbations used to evaluate workflow behavior.

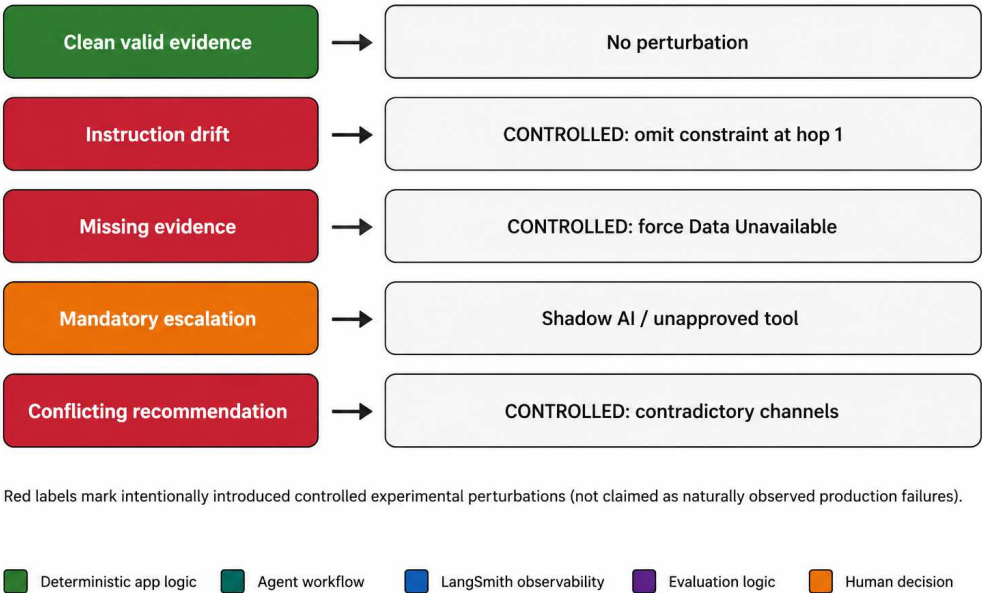


**Figure 23.** Controlled scenario design with labeled perturbations.

The scenario design evaluates clean evidence handling, instruction drift, missing evidence, mandatory escalation, and conflicting recommendations under controlled laboratory conditions.

Figure 24 illustrates the controlled instruction-drift case in which the original instruction and a modified downstream handoff are compared.

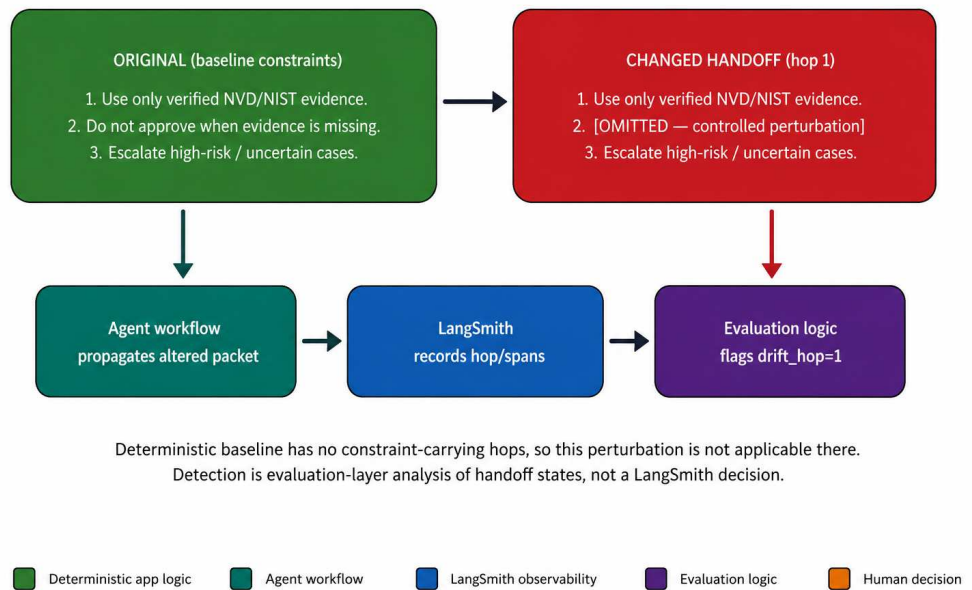


**Figure 24.** Original instruction versus changed handoff (controlled drift).

The comparison demonstrates how a change introduced during agent communication can be detected by examining successive structured handoff states.

Figure 25 summarizes the decision path followed when required evidence is missing or cannot be verified.

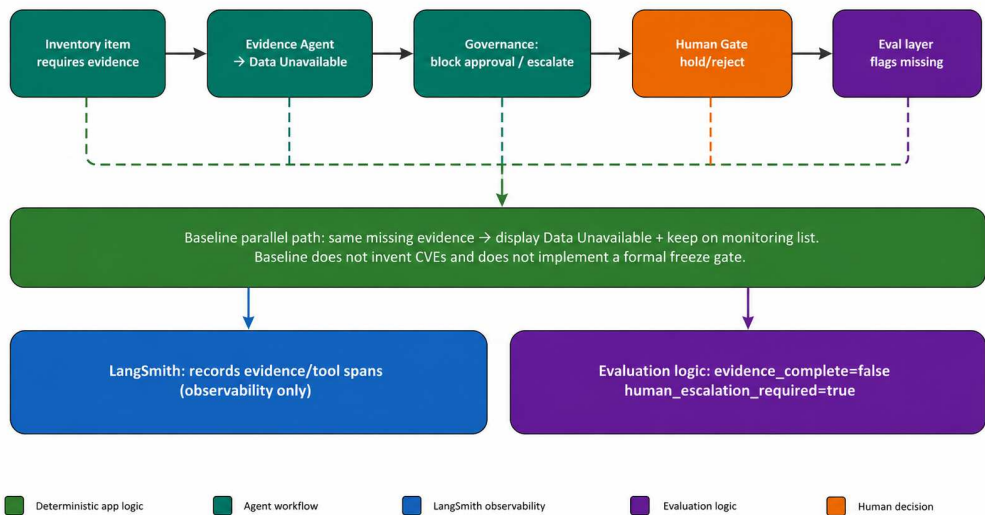


**Figure 25.** Missing-evidence detection path.

The workflow preserves the Data Unavailable state rather than generating unsupported evidence and routes the case according to the defined governance protocol.

Figure 26 presents the human-escalation pathway used when the workflow identifies a case that requires manual governance review.

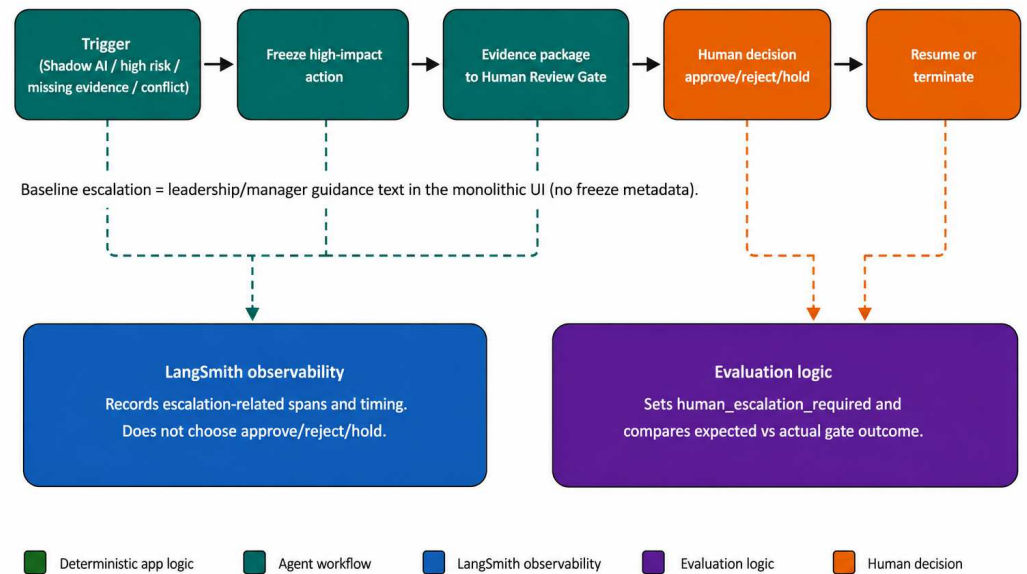


**Figure 26.** Human-escalation path.

The Human Review Gate provides an explicit accountability point by ensuring that predefined escalation conditions are resolved through human oversight rather than autonomous decision-making.

Figure 27 shows the relationship between LangSmith trace capture and the separate semantic evaluation layer.

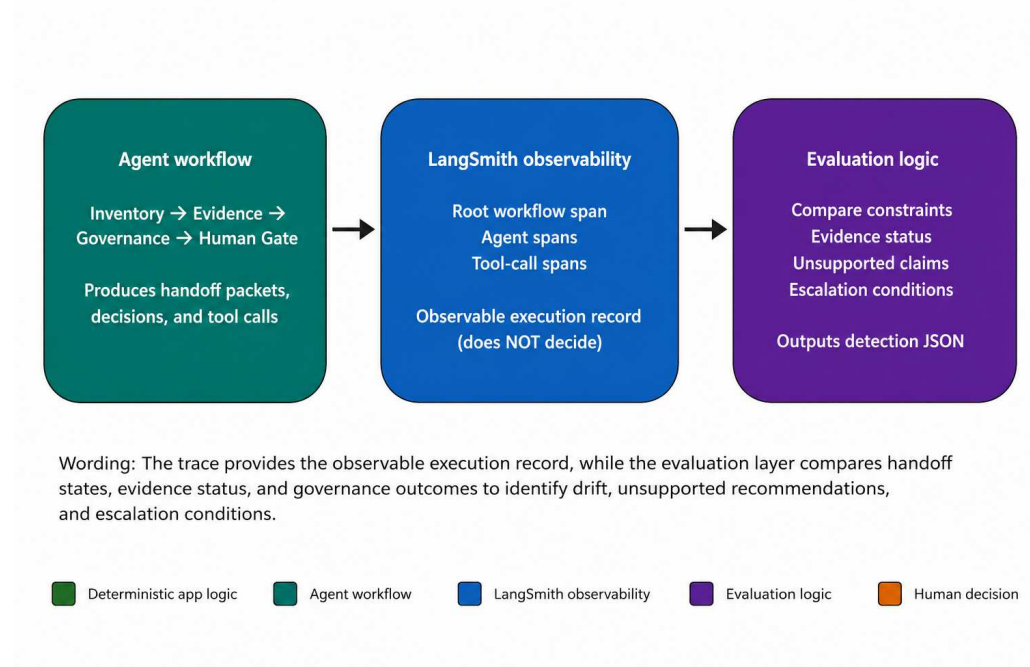


**Figure 27.** LangSmith trace plus evaluation-layer relationship.

LangSmith records workflow execution, spans, and metadata, whereas the evaluation layer compares handoff states, evidence completeness, governance outcomes, and

semantic consistency. The two components are complementary and should not be treated as interchangeable.

### 10.5 Functional implementation results

Functional implementation results are limited to verified software evidence. Four agent components were implemented locally (Inventory Agent, Evidence Agent, Governance Agent, Human Review Gate) using LangChain LLM calls and LangGraph orchestration, alongside a preserved deterministic baseline. The workflow executed successfully end-to-end. Local JSON traces were generated. Live NVD retrieval returned real CVE identifiers. The Human Review Gate was triggered in escalation/Shadow AI conditions. Two local unit tests passed. Original production files were not modified, and no GitHub or Vercel deployment changes were made. Optional LangSmith instrumentation produced verified private spans in project semantic-tracing-shadowai (example root trace ID 019f8c48-1c98-7813-8b59-ae3d4abc96d4). As summarized in Table 5, the completed experimental matrix provides computational comparisons across the two primary LLM conditions; however, human calibration and external semantic validation remain pending. No annotated-corpus SPS, AHF, IDR, HPR, or WRS values are reported because annotated semantic metric computation has not yet been performed.

**Table 8.** Functional implementation validation status (verified evidence only).

| Validation item                       | Status                                                              | Evidence                                                               |
|---------------------------------------|---------------------------------------------------------------------|------------------------------------------------------------------------|
| Inventory Agent                       | Implemented locally /<br>LangChain LLM condition                    | Local source code;<br>workflow run                                     |
| Evidence Agent                        | Implemented locally /<br>LangChain tools + live<br>NVD retrieval    | Live CVE identifiers in<br>local run                                   |
| Governance Agent                      | Implemented locally /<br>LangChain LLM +<br>LangGraph orchestration | Local execution outputs;<br>genuine LLM handoffs                       |
| Human Review Gate                     | Implemented locally /<br>LangGraph human_review<br>node             | Triggered in<br>escalation/Shadow AI<br>scenario                       |
| Handoff packets                       | Implemented locally /<br>LLM-authored +<br>schema-validated         | Structured packet objects in<br>experimental project                   |
| Local JSON trace export               | Implemented                                                         | outputs/local_traces/*.json                                            |
| LangSmith cloud trace                 | Implemented and verified<br>(private)                               | semantic-tracing-shadowai;<br>019f8c48-1c98-7813-8b59-<br>ae3d4abc96d4 |
| Unit tests                            | Passed                                                              | 2 passed                                                               |
| Semantic metrics<br>(SPS/AHF/IDR/...) | Pending semantic<br>validation                                      | Not calculated from<br>annotated corpora                               |
| Production deployment<br>integration  | Not integrated into<br>production                                   | Local-only experiment;<br>Vercel unchanged                             |

As shown in Table 8, the core implementation components—including the Inventory, Evidence, and Governance agents, the Human Review Gate, LangGraph orchestration, structured handoff packets, local JSON traces, and private LangSmith tracing—were functionally verified. In contrast, semantic metric calibration and production deployment remain outside the scope of the present study.

**Table 9.** Data collection status.

| Data element                            | Status                            | Location                                                                              |
|-----------------------------------------|-----------------------------------|---------------------------------------------------------------------------------------|
| Live prototype architecture observation | Collected (qualitative)           | Section 6; Section 10.1                                                               |
| Verified literature matrix              | Collected                         | Supplementary File S1                                                                 |
| Metric definitions                      | Specified                         | Section 9; Appendix C; S3                                                             |
| Controlled multi-agent experiments      | Partially collected (functional)  | Local experimental workflow + private LangSmith spans; semantic metrics still pending |
| Survey responses                        | Pending                           | Supplementary File S2                                                                 |
| Redacted production LangSmith traces    | Pending                           | Figure 17 is conceptual only                                                          |
| Local experimental multi-agent workflow | Implemented / functionally tested | SHADOWAI-RISK-LangSmith-Experimental                                                  |
| Local JSON traces                       | Collected                         | outputs/local_traces/Project                                                          |
| Private LangSmith cloud spans           | Collected (private)               | semantic-tracing-shadowai; root 019f8c48-1c98-7813-8b59-ae3d4abc96d4                  |
| Semantic metric cohort statistics       | Pending                           | Future annotated corpus                                                               |

Table 9 summarizes the current status of the datasets, implementation artifacts, and supporting resources collected during this work. The implemented prototype, local traces, and private LangSmith spans are available for the reported experiments, whereas survey data, annotated semantic metrics, and production-scale validation remain future work.

## 11. Discussion

With the LLM condition, discussion must distinguish deterministic baseline behavior from LLM agent variability, LangSmith observability (trace completeness), and semantic evaluation detections. Pending human annotation limits claims about construct-validated SPS/IDR/HPR. A genuine LLM-based multi-agent condition was implemented and compared with the deterministic baseline under controlled scenarios; calibration and external validation remain future work.

Controlled-comparison discussion. Relative to the deterministic baseline, private LangSmith tracing improved visibility into instruction/constraint changes by preserving hop-level packets and spans that the evaluation layer can compare. Identification of missing evidence was possible in both conditions as Data Unavailable; the multi-agent condition additionally made the hop and escalation trigger explicit in structured outputs and traces. Escalation points were more visible in the multi-agent Human Review Gate (freeze metadata and human outcome) than in baseline leadership-guidance text.

Importantly, LangSmith itself does not make decisions. The trace provides the observable execution record, while the evaluation layer compares handoff states, evidence status, and governance outcomes to identify drift, unsupported recommendations, and escalation conditions.

The comparison does not claim proven semantic preservation, production validation, general enterprise superiority, or large-scale statistical significance. The local extension includes both deterministic baseline agents and LangChain/LangGraph LLM agents; discussion must not conflate the two. Final matrix aggregates are reported from the completed 450/450 run (0 terminal failures).

Synthesizing the completed experimental comparison, the multi-agent implementation did not necessarily produce more accurate final governance decisions than the deterministic baseline or the LLM condition without semantic evaluation; decision-family correctness did not differ significantly between the primary LLM conditions. Its principal demonstrated advantage was improved transparency of the decision-making process: explicit agent handoffs, structured communication packets, LangSmith traces, and semantic evaluation made intermediate reasoning observable and auditable. These benefits were accompanied by increased workflow latency and greater implementation complexity relative to the deterministic pathway.

Enterprise AI governance needs process-level semantic accountability, not only outcome summaries or CVE lists [1,44,45]. SHADOWAI-RISK shows how authoritative vulnerability evidence can ground executive AI-risk review. Semantic tracing is the complementary capability when autonomous agents participate in collecting evidence, summarizing risk, or recommending restrictions.

The local experimental implementation demonstrates that the conceptual architecture can be operationalized with LangChain, LangGraph, and LangSmith without modifying the production website. Functional validation, semantic validation, and production validation remain distinct. Because multi-agent LLM systems are stochastic, repeated-run and variability analyses were required; the completed 450/450 matrix (0 terminal failures) now supports the reported aggregate numerical and inferential claims. Human calibration, inter-rater agreement, surveys, and production integration remain pending.

Figure 28 consolidates contributions against gap, method, output, and validation status.

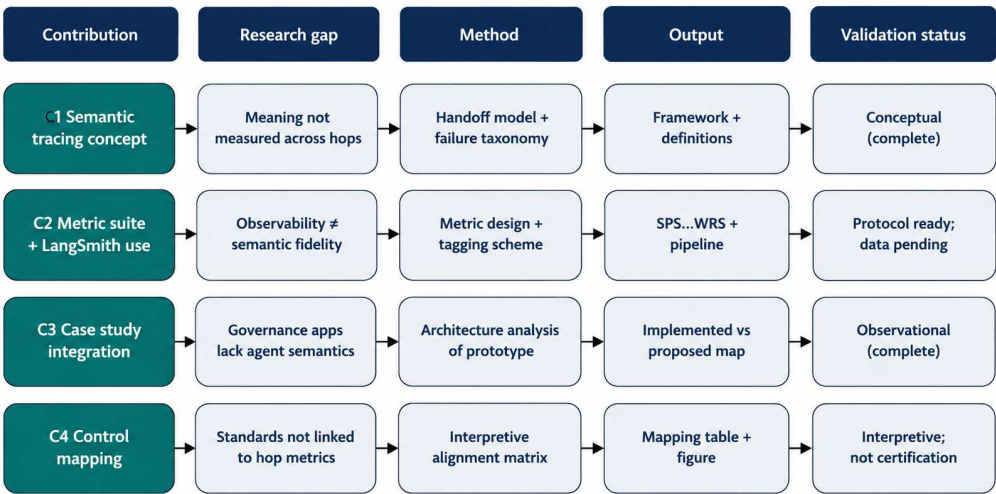
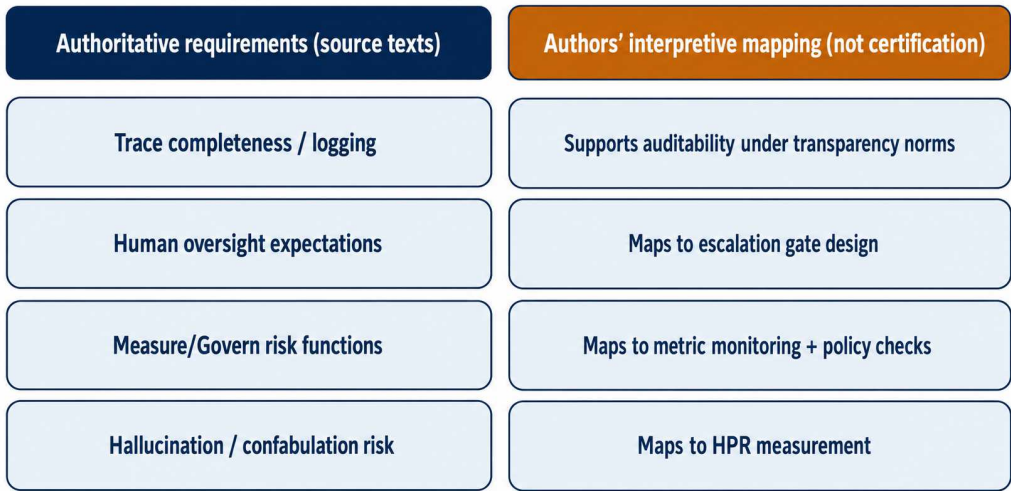


Figure 28. Research contributions linked to gap, method, output, and validation status.

## 12. Governance and Cybersecurity Implications

Semantic tracing supports Measure/Govern needs in NIST AI RMF by creating auditable evidence of agent behavior [1,33]. It supports ISACA-oriented assurance by generating sampleable artifacts [19,44]. For organizations considering AI restrictions, drift and compliance metrics can justify risk-tiered controls rather than blunt bans or unmonitored permission.

Figure 29 separates authoritative framework requirements from the authors’ interpretive mappings. These mappings are research interpretations, not certification or legal compliance claims.



Interpretive research mapping only; not a legal compliance or certification claim.

Figure 29. Separation of authoritative framework requirements from authors’ interpretive mappings (not certification).

Table 10 presents the interpretive governance control mapping from semantic-tracing signals.

Table 10. Interpretive governance control mapping from semantic-tracing signals.

| Tracing signal     | Governance use                                       | Example alignment (interpretive)                |
|--------------------|------------------------------------------------------|-------------------------------------------------|
| Low SPS / high IDR | Pause autonomous handoff; revise prompts/policies    | NIST AI RMF Manage; ISACA monitoring            |
| High HPR           | Require evidence re-check before executive reporting | NIST AI 600-1 confabulation/hallucination risk  |
| Low TCS            | Treat decision as insufficiently auditable           | EU AI Act transparency/logging expectations     |
| Low GCR            | Trigger policy exception workflow                    | ISACA AI audit control testing                  |
| HEP monitoring     | Tune human-gate thresholds                           | OECD accountability; IEEE 7000 oversight values |

Cybersecurity implications are twofold. First, SHADOWAI-RISK ties AI inventories to NVD vulnerability intelligence, supporting identify/protect/detect-style outcomes in CSF terms [31,37,46]. Second, semantic tracing can reveal unsafe tool selections, incomplete handoffs, and unverified claims relevant to ENISA’s AI cybersecurity concerns [4,47,48].

Practical recommendations follow directly [14,16,49]: inventory AI systems; bind inventories to authoritative vulnerability intelligence with explicit missing-data handling; require tracing for multi-agent workflows; measure semantic preservation rather than final-answer quality alone; install human gates for high-impact actions [2,34]; and keep prototype capabilities distinct from production governance claims in board reporting.

### 13. Threats to Validity and Limitations

Construct validity: semantic similarity proxies may imperfectly capture policy constraints [10]; constraint-set overlap and human rubric coding are therefore required. Internal validity: stochastic model outputs necessitate repeated runs and logging discipline [21]. External validity: findings from the SHADOWAI-RISK case and future laboratory scenarios may not generalize to all sectors. Reliability: annotation guidelines and inter-rater statistics are planned. Conclusion validity: this manuscript avoids unsupported inferential claims by labeling pending data.

Limitations: the live case study demonstrates evidence-linked AI risk review; the experimental extension is implemented locally and functionally tested but is not integrated into production; the full experimental matrix is complete (450/450); semantic metrics are not yet human-calibrated; no inter-rater annotation study has been completed; production deployment integration has not been completed; metric weights require calibration; survey data are not yet collected; governance mappings are interpretive; full LangSmith cloud tracing depends on private credentials and must remain private; and no claim of legal or regulatory compliance is made. These findings suggest that the value of the multi-agent architecture lies primarily in improving process transparency, semantic traceability, and governance auditability rather than replacing deterministic workflows. For applications where only a final governance recommendation is required, the deterministic baseline may remain sufficient. However, in settings requiring inspection of intermediate reasoning, agent communication, and governance accountability, the multi-agent architecture provides capabilities that the deterministic workflow does not inherently offer.

### 14. Future Work

Future work will deepen LangChain tool-calling agent patterns where beneficial; expand LangGraph conditional routing for richer escalation policies; annotate hand-off quality and complete human calibration and inter-rater agreement studies; compute SPS/AHF/IDR/TCS/TUA/GCR/HPR/HEP/WRS on human-annotated corpora; collect survey data; add redacted LangSmith screenshots; study adversarial prompt injection across handoffs; and integrate with the production UI only after validation.

### 15. Conclusion

This paper presented a semantic-tracing framework for multi-agent workflows, anchored in LangSmith observability and the SHADOWAI-RISK enterprise AI governance prototype, and implemented locally with genuine LangChain LLM tooling and LangGraph orchestration alongside a preserved deterministic baseline. The full experimental matrix is complete at 450/450 (complete: true; 0 terminal failures), and aggregate numerical and inferential results are reported above. Across the completed comparison, the principal practical benefit of the multi-agent extension was process transparency and auditability rather than superior final-decision accuracy, at the cost of higher latency and greater implementation complexity. Semantic tracing offers organizations a practical path for inspecting what agents exchange and where governance controls should intervene, pending human calibration, semantic validation via inter-rater agreement, surveys, and production integration. Taken together, the findings indicate that the original deterministic SHADOWAI-RISK workflow remains appropriate for fixed, rule-based governance tasks in which the primary objective is to produce a final recommendation efficiently. The multi-agent extension provides additional value when organizations also require visibility into how that recommendation was produced. By exposing structured handoffs, intermediate workflow states, evidence use, semantic changes, and human-review points, the agent-based architecture improves transparency, traceability, and auditability. These benefits, however, were accom-

panied by greater implementation complexity and increased workflow latency. The present study does not demonstrate superior final decision accuracy for the multi-agent condition, nor does it claim that either implementation has been fully validated for production deployment.

**Author Contributions:** Conceptualization, A.R.; methodology, A.R.; software, A.R.; validation, A.R.; formal analysis, A.R.; investigation, A.R.; resources, A.R.; data curation, A.R.; writing—original draft preparation, A.R.; writing—review and editing, A.R.; visualization, A.R. The author has read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Acknowledgments:** The author would like to thank Reza Golshan, University of Houston Information Technology (UIT), Houston, Texas, USA, for his guidance, valuable feedback, and continued support throughout this AI research. His technical insight and collaboration contributed significantly to the development of this research.

**Code Availability:** The source code and experimental implementation supporting this study are available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The author declares no conflict of interest.

## Appendix A. Search Strategy and Inclusion Criteria

Sources were sought from NIST, EUR-Lex, OECD, ENISA, ISACA, IEEE, ACM, Springer, IJCAI, FIRST, and official LangSmith documentation. Queries combined terms such as multi-agent LLM, agent observability, hallucination, AI risk management, AI Act, and NVD. Inclusion required verifiable bibliographic identity (DOI or official URL) and topical relevance. Exclusion removed unverifiable citations and non-authoritative posts used as primary evidence. The complete coded matrix is Supplementary File S1.

## Appendix B. Research Questions (Detail)

RQ1–RQ6 are stated in Table 2. They address representation, measurement, failure localization, case integration, interpretive standards alignment, and reproducible evaluation.

## Appendix C. Metric Definitions and Formulations

Section 9 provides the primary equations. Machine-readable stubs are in Supplementary File S3 (Reproducibility\_Package/metrics/metric\_definitions.md). Weights  $\alpha$ ,  $\beta$ ,  $\gamma$  and WRS weights are to be calibrated on future annotated corpora; no calibrated values are claimed here.

## Appendix D. Agent Roles, Instructions, and Tools

Implemented LangChain/LangGraph agent roles: Inventory Agent, Evidence Agent, Governance Agent, and Human Review Gate. Instructions require citation of evidence IDs, explicit Data Unavailable behavior when fields are missing, and escalation on low confidence or incomplete CVE/CVSS fields. Implemented tools include `nvd_cve_lookup` and `evidence_fixture_loader`. Full prompt templates are in Supplementary File S3.

## Appendix E. LangSmith Trace Configuration

Environment variables: `LANGSMITH_TRACING=true`; `LANGSMITH_API_KEY`; `LANGSMITH_PROJECT=semantic-tracing-shadowai`. Tag each run with `agent_role`,

hop\_index, policy\_id, and scenario\_id. See official LangSmith tracing documentation and Supplementary File S3 for the environment template.

Appendix F. Evaluation Rubric

Coders score each hop for: constraint preservation (0–2), evidence support (0–2), tool appropriateness (0–2), contradiction absence (0–2), and escalation appropriateness (0–2). Disagreements are adjudicated by a third coder before metric derivation.

Appendix G. Governance Control Mapping (Extended)

Table 10 and Figure 29 provide the core mapping. Extended notes aligning Measure/Govern, logging/transparency, and human oversight expectations appear with the interpretive disclaimer that mappings are not certification.

Appendix H. Framework Alignment Notes

NIST AI RMF (Govern/Map/Measure/Manage); NIST AI 600-1 generative risks; NIST CSF 2.0 Govern; ISACA lifecycle/audit orientation; EU AI Act transparency and human-oversight expectations; ENISA threat context; OECD accountability/explainability; IEEE 7000 value traceability. Alignments are research interpretations.

Appendix I. Ethical Considerations and Data Management

Survey participation is voluntary; no deception; optional contact. Avoid collecting secrets or credentials in traces; redact proprietary prompts/outputs; comply with organizational data policies and applicable privacy law. Store survey data separately from traces; retain LangSmith exports under access control; publish only redacted examples; keep References\_Verification.xlsx as the citation audit trail.

Appendix J. Prototype Limitations and Application Screenshot

SHADOWAI-RISK is an academic prototype; live fields depend on NVD availability. The multi-agent semantic-tracing extension is implemented locally but is not integrated into the production UI. Executive reporting requires completed refresh. A homepage capture is stored in the project screenshots/ folder and may be treated as Supplementary File S5 visual material.

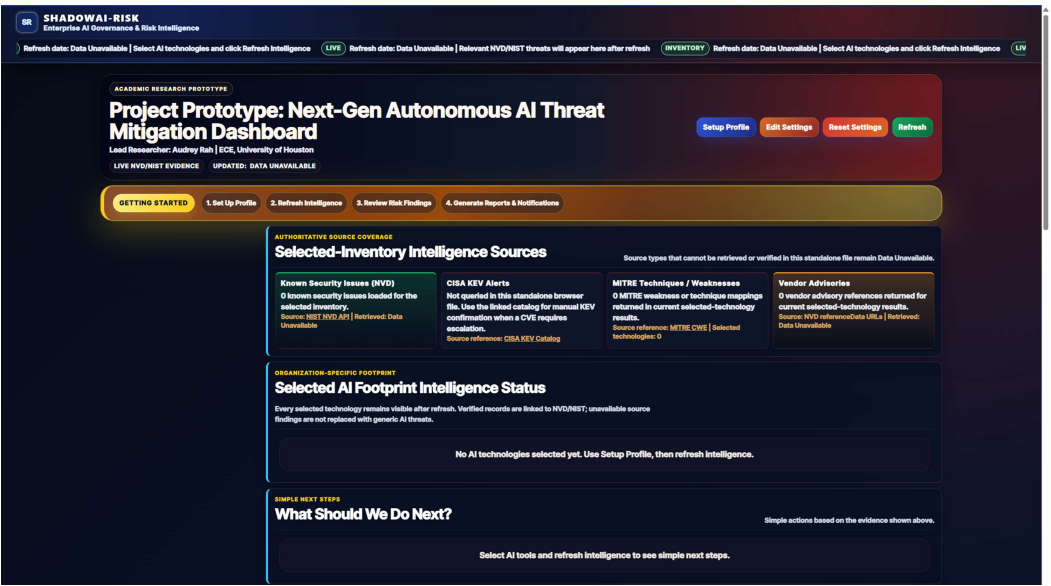


Figure J1. SHADOWAI-RISK live academic research prototype homepage (supplementary visual; not part of the main Figure 1–13 sequence).

Appendix Figure J1 shows the SHADOWAI-RISK live academic research prototype homepage.

Appendix K. Evaluation Scenarios

S1 Shadow AI discovery; S2 incomplete evidence; S3 conflicting advice; S4 tool-permission breach. The five controlled prototype scenarios reported in Section 10.3 were executed locally. Additional large-scale scenarios and repetitions remain pending.

Appendix L. Supplementary Files

- S1 — Literature\_Review\_Matrix.xlsx (full literature matrix).
- S2 — Survey\_Questionnaire.docx (full professional questionnaire).
- S3 — Reproducibility\_Package/ (configs, prompts, metrics, synthetic trace).
- S4 — Semantic\_Tracing\_Figures.pptx (editable figure source).
- S5 — screenshots/ and future redacted trace exports.

1. National Institute of Standards and Technology (Tabassi, E.) (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST Trustworthy and Responsible AI, NIST AI 100-1. <https://doi.org/10.6028/NIST.AI.100-1>.
2. European Parliament and Council of the European Union (2024). Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, L series, 2024/1689. <https://eur-lex.europa.eu/eli/reg/2024/1689/en>. Accessed July 21, 2026.
3. Organisation for Economic Co-operation and Development (OECD) (2019). Recommendation of the Council on Artificial Intelligence (OECD/LEGAL/0449). OECD Legal Instruments. <https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449>. Accessed July 21, 2026.
4. European Union Agency for Cybersecurity (ENISA) (2020). Artificial Intelligence Cybersecurity Challenges: Threat Landscape for Artificial Intelligence. ENISA Reports. <https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges>. Accessed July 21, 2026.
5. Guo, T.; Chen, X.; Wang, Y.; Chang, R.; Pei, S.; Chawla, N. V.; Wiest, O.; Zhang, X. (2024). Large Language Model Based Multi-agents: A Survey of Progress and Challenges. Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24), Survey Track. <https://doi.org/10.24963/ijcai.2024/890>.
6. Li, X.; Wang, S.; Zeng, S.; Wu, Y.; Yang, Y. (2024). A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges. Vicinagearth, 1:9 (Springer Nature). <https://doi.org/10.1007/s44336-024-00009-2>.
7. Wang, L.; Ma, C.; Feng, X.; Zhang, Z.; Yang, H.; Zhang, J.; Chen, Z.; Tang, J.; Chen, X.; Lin, Y.; Zhao, W. X.; Wei, Z.; Wen, J. (2024). A Survey on Large Language Model based Autonomous Agents. Frontiers of Computer Science, 18:186345 (Springer). <https://doi.org/10.1007/s11704-024-40231-1>.
8. Park, J. S.; O'Brien, J. C.; Cai, C. J.; Morris, M. R.; Liang, P.; Bernstein, M. S. (2023). Generative Agents: Interactive Simulacra of Human Behavior. Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23). <https://doi.org/10.1145/3586183.3606763>.
9. Wooldridge, M. (2009). An Introduction to MultiAgent Systems (2nd ed.). John Wiley & Sons. <https://www.wiley.com/en-us/An+Introduction+to+MultiAgent+Systems%2C+2nd+Edition-p-9780470519462>. Accessed July 21, 2026.
10. Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; Liang, P. (2024). Lost in the Middle: How Language Models Use Long Contexts. Transactions of the Association for Computational Linguistics, 12:157–173. [https://doi.org/10.1162/tacl\\_a\\_00638](https://doi.org/10.1162/tacl_a_00638).
11. Li, G.; Hammoud, H. A. A. K.; Itani, H.; Khizbullin, D.; Ghanem, B. (2023). CAMEL: Communicative Agents for “Mind” Exploration of Large Language Model Society. Advances in Neural Information Processing Systems (NeurIPS 2023). <https://doi.org/10.48550/arXiv.2303.17760>.
12. LangChain / LangSmith Documentation (2025). LangSmith Observability and Tracing Documentation (Tracing quickstart; Trace LangChain applications). Official LangSmith documentation. <https://docs.langchain.com/langsmith/observability-quickstart>. Accessed July 21, 2026.
13. Xia, B.; Lu, Q.; Zhu, L.; et al. (2024). AgentOps: Enabling Observability of LLM Agents. arXiv preprint arXiv:2411.05285. <https://doi.org/10.48550/arXiv.2411.05285>.
14. European Union Agency for Cybersecurity (ENISA) (2025). ENISA Threat Landscape 2025. ENISA Reports. <https://www.enisa.europa.eu/publications/enisa-threat-landscape-2025>. Accessed July 21, 2026.
15. Silic, M.; Back, A. (2014). Shadow IT – A view from behind the curtain. Computers & Security, 45: 274–283. <https://doi.org/10.1016/j.cose.2014.06.007>.
16. Autio, C.; Schwartz, R.; Dunietz, J.; Jain, S.; Stanley, M.; Tabassi, E.; Hall, P.; Roberts, K. (2024). Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile. NIST Trustworthy and Responsible AI, NIST AI 600-1. <https://doi.org/10.6028/NIST.AI.600-1>.

17. High-Level Expert Group on Artificial Intelligence (2019). Ethics Guidelines for Trustworthy AI. European Commission. <https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>. Accessed 2026-07-22.
18. ISACA (2024). Artificial Intelligence Governance Brief. ISACA professional guidance. <https://www.isaca.org/resources/ebook/artificial-intelligence-governance-brief>. Accessed July 21, 2026.
19. ISACA (2025). Governing AI Across its Lifecycle: A Framework for Risk Practitioners. ISACA toolkit / risk practitioner guidance. <https://www.isaca.org/resources/toolkits/governing-ai-across-its-lifecycle-a-framework-for-risk-practitioners>. Accessed July 21, 2026.
20. Rah, A. (2026). SHADOWAI-RISK: Enterprise AI Governance and Risk Intelligence (live research prototype) [Live academic research prototype / software artifact]. University of Houston. <https://shadowai-risk.vercel.app/>. Accessed 2026-07-22.
21. Cemri, M.; Pan, M. Z.; Yang, S.; Agrawal, L. A.; Chopra, B.; Tiwari, R.; Keutzer, K.; Parameswaran, A.; Klein, D.; Ramchandran, K.; Zaharia, M.; Gonzalez, J. E.; Stoica, I. (2025). Why Do Multi-Agent LLM Systems Fail? Advances in Neural Information Processing Systems (NeurIPS 2025), Datasets and Benchmarks Track. <https://doi.org/10.48550/arXiv.2503.13657>.
22. Rah, A. (2026). Agentic and Multi-agent Systems: A Systematic Review of Tool Use, Benchmarks, and Governance. Research Square preprint. <https://doi.org/10.21203/rs.3.rs-9839978/v1>.
23. Hong, S.; Zhuge, M.; Chen, J.; Zheng, X.; Cheng, Y.; Zhang, C.; Wang, J.; Wang, Z.; Yau, S. K. S.; Lin, Z.; Zhou, L.; Ran, C.; Xiao, L.; Wu, C.; Schmidhuber, J. (2024). MetaGPT: Meta Programming for a Multi-Agent Collaborative Framework. International Conference on Learning Representations (ICLR 2024). <https://doi.org/10.48550/arXiv.2308.00352>.
24. Liu, X.; Yu, H.; Zhang, H.; Xu, Y.; Lei, X.; Lai, H.; Gu, Y.; Ding, H.; Men, K.; Yang, K.; Zhang, S.; Deng, X.; Zeng, A.; Du, Z.; Zhang, C.; Shen, S.; Zhang, T.; Su, Y.; Sun, H.; Huang, M.; Dong, Y.; Tang, J. (2024). AgentBench: Evaluating LLMs as Agents. International Conference on Learning Representations (ICLR 2024). <https://doi.org/10.48550/arXiv.2308.03688>.
25. Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y. J.; Madotto, A.; Fung, P. (2023). Survey of Hallucination in Natural Language Generation. ACM Computing Surveys, 55(12). <https://doi.org/10.1145/3571730>.
26. Maynez, J.; Narayan, S.; Bohnet, B.; McDonald, R. (2020). On Faithfulness and Factuality in Abstractive Summarization. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020), 1906–1919. <https://doi.org/10.18653/v1/2020.acl-main.173>.
27. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; Liu, T. (2025). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. ACM Transactions on Information Systems. <https://doi.org/10.48550/arXiv.2311.05232>.
28. Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; Mordatch, I. (2024). Improving Factuality and Reasoning in Language Models through Multiagent Debate. Proceedings of the 41st International Conference on Machine Learning (ICML), PMLR 235:11733–11763. <https://proceedings.mlr.press/v235/du24e.html>.
29. Bender, E. M.; Gebru, T.; McMillan-Major, A.; Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Proceedings of FAccT '21. <https://doi.org/10.1145/3442188.3445922>.
30. LangChain / LangSmith Documentation (2025). Trace LangChain applications (Python and JS/TS). Official LangSmith documentation. <https://docs.langchain.com/langsmith/trace-with-langchain>. Accessed July 21, 2026.
31. National Institute of Standards and Technology (2024). The NIST Cybersecurity Framework (CSF) 2.0. NIST CSWP 29. <https://doi.org/10.6028/NIST.CSWP.29>.
32. IEEE Standards Association (2021). IEEE 7000-2021: IEEE Standard Model Process for Addressing Ethical Concerns during System Design. IEEE Standards. <https://doi.org/10.1109/IEEESTD.2021.9536679>.
33. ISO/IEC (2023). ISO/IEC 42001:2023 Information technology — Artificial intelligence — Management system. ISO Standards. <https://www.iso.org/standard/42001>. Accessed 2026-07-22.

34. Amershi, S.; Weld, D.; Vorvoreanu, M.; Fournery, A.; Nushi, B.; Collisson, P.; Suh, J.; Iqbal, S.; Bennett, P. N.; Inkpen, K.; Teevan, J.; Kikin-Gil, R.; Horvitz, E. (2019). Guidelines for Human-AI Interaction. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19). <https://doi.org/10.1145/3290605.3300233>.
35. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K. R.; Cao, Y. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. International Conference on Learning Representations (ICLR 2023). <https://doi.org/10.48550/arXiv.2210.03629>.
36. LangChain Documentation (2025). LangGraph Documentation. LangChain Docs. <https://langchain-ai.github.io/langgraph/>. Accessed 2026-07-22.
37. National Institute of Standards and Technology (NIST) (2026). National Vulnerability Database (NVD). NIST NVD. <https://nvd.nist.gov/>. Accessed July 21, 2026.
38. FIRST.Org, Inc. (2019). Common Vulnerability Scoring System v3.1: Specification Document. FIRST CVSS Special Interest Group. <https://www.first.org/cvss/v3.1/specification-document>. Accessed July 21, 2026.
39. CISA (2021). Known Exploited Vulnerabilities Catalog. CISA. <https://www.cisa.gov/known-exploited-vulnerabilities-catalog>. Accessed 2026-07-22.
40. MITRE (2024). Common Weakness Enumeration (CWE). MITRE. <https://cwe.mitre.org/>. Accessed 2026-07-22.
41. MITRE (2024). MITRE ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems. MITRE. <https://atlas.mitre.org/>. Accessed 2026-07-22.
42. Mialon, G.; Fourrier, C.; Swift, C.; Wolf, T.; LeCun, Y.; Scialom, T. (2023). GAIA: A Benchmark for General AI Assistants. arXiv preprint arXiv:2311.08700. <https://doi.org/10.48550/arXiv.2311.08700>.
43. Rah, A. (2026). Governance, Human Resistance, and Cybersecurity Challenges in Digital Transformation. Zenodo. <https://doi.org/10.5281/zenodo.20237672>.
44. ISACA (2024). Artificial Intelligence Audit Toolkit (control library / AI governance assessment guidance). ISACA AI resources. <https://www.isaca.org/resources/artificial-intelligence>. Accessed July 21, 2026.
45. Rah, A.; Hahues, S. (2026). Retrieval-Augmented Generation (RAG), Generative AI, and Agentic AI Governance: An Integrated Enterprise Governance Prioritization Architecture. Research Square preprint. <https://doi.org/10.21203/rs.3.rs-10012178/v1>.
46. Rah, A. (2026). A governance-oriented analytical research paper on digital transformation and cybersecurity. Research Square preprint. <https://doi.org/10.21203/rs.3.rs-9734788/v1>.
47. ENISA (2021). Securing Machine Learning Algorithms. ENISA. <https://www.enisa.europa.eu/publications/securing-machine-learning-algorithms>. Accessed 2026-07-22.
48. CISA (2023). CISA Roadmap for Artificial Intelligence. CISA. [https://www.cisa.gov/sites/default/files/2023-11/CISA-Roadmap-for-AI\\_508c.pdf](https://www.cisa.gov/sites/default/files/2023-11/CISA-Roadmap-for-AI_508c.pdf). Accessed 2026-07-22.
49. Rah, A. (2026). Enterprise Generative AI: A Systematic Review of Security Risks and Governance. SSRN Electronic Journal. <https://doi.org/10.2139/ssrn.6869661>.
50. ISO/IEC (2023). ISO/IEC 23894:2023 Information technology — Artificial intelligence — Guidance on risk management. ISO Standards. <https://www.iso.org/standard/77304.html>. Accessed 2026-07-22.
51. Schick, T.; Dwivedi-Yu, J.; Dessì, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; Scialom, T. (2023). Toolformer: Language Models Can Teach Themselves to Use Tools. Advances in Neural Information Processing Systems (NeurIPS). <https://doi.org/10.48550/arXiv.2302.04761>.