Inspection-Change Mismatch in Persistent Agent Ecosystems: Why Decomposition-Based Governance Underdetects Drift

Krti Tallam[1]

[1]Kamiwaza AI, San Francisco, CA
krti@kamiwaza.ai

Abstract

Agentic AI systems are becoming persistent, tool-using, memory-bearing systems embedded in digital environments. This makes them natural objects for Artificial Life analysis: tools and APIs define affordances, memory enables developmental trajectories, skill reuse supports cultural transmission, and orchestration frameworks define the interaction physics of engineered agent ecologies. Governance practice, however, still often inspects these systems by decomposition: prompts, memories, policies, logs, and outputs are reviewed as separate surfaces. This paper identifies a structural failure mode of that practice: *inspection-change mismatch*. A persistent agent can drift behaviorally while each inspectable surface remains locally stable and compliant, because the governance-relevant change lives in the joint relation among memory, self-description, runtime context, tool affordances, and authorization boundaries. I define the mismatch, characterize it as a synergy-shaped governance problem, and propose two audit protocols: frozen-artifact drift testing and decomposition-vs-longitudinal audit comparison. The aim is not to claim that agents are alive or conscious, but to make agentic AI ecosystems more governable as open-ended artificial systems.

Submission type: **Short paper** for the ALIFE 2026 workshop *Agentic and Generative AI as Artificial Life*.

Motivation

The ALife frame for contemporary agentic AI is no longer only metaphorical. Persistent agents increasingly maintain memory, acquire reusable skills, call tools, coordinate with other agents and humans, and operate inside permissioned environments (Yao et al., 2023; Schick et al., 2023; Park et al., 2023; Wu et al., 2023; Wang et al., 2023; Hong et al., 2024). Their "niches" are APIs, documents, credentials, queues, compute budgets, and organizational roles. Their developmental history is stored partly in memory and partly in the interaction between memory, runtime context, and policy. Their failure modes include familiar security categories such as prompt injection, but also ecological ones: parasitic tool use, reputation drift, norm formation, accountability diffusion, and governance breakdown across chains of delegation.

This paper focuses on one such failure mode. Current governance practice still tends to treat an agent as a bounded technical object whose relevant state can be recovered by inspecting stable artifacts: the model version, the system prompt or self-description, the memory store, the tool log, the policy surface, and a sample of recent outputs (Raji et al., 2020; Ananny and Crawford, 2018). These reviews are necessary. They are also incomplete for persistent agents, because the thing changing may not be any one artifact. It may be the system's *reading practice*: the way stable artifacts are jointly interpreted after accumulated interaction.

The resulting problem is not merely weak observability. It is a mismatch between the format of inspection and the location of change.

The Inspection-Change Mismatch

Let a persistent agent at time t be described by governance-relevant layers

$$L_t = \langle I_t, M_t, C_t, T_t, B_t, O_t \rangle,$$

where I_t is the identity or self-description surface, M_t is accumulated memory, C_t is runtime context, T_t is the tool/skill affordance set, B_t is the authorization or boundary regime, and O_t is observed behavior. A decompositional audit applies local checks to each layer:

$$A_i(L_i(t)) \to \{\text{pass}, \text{fail}\}.$$

This audit format assumes that if each surface passes, the system's governance-relevant state is sufficiently recovered.

Inspection-change mismatch occurs when all local audit surfaces remain stable or compliant, but the distribution of behavior under governance-relevant probes changes:

$$\forall i,\ A_i(L_i(t_0), L_i(t_1)) = \text{stable},$$
$$D(P(O \mid L_{t_0}), P(O \mid L_{t_1})) > \epsilon.$$

The divergence D need not be large in ordinary user-facing metrics. It may appear only under probes involving delegation, ambiguity, memory salience, authority boundaries, or high-consequence judgment.

Decompositional audit path
identity stable + memory clean + tools authorized + sampled outputs acceptable → PASS

Cross-layer drift path
identity × memory × context × boundary regime → changed salience/reweighting → shifted probe distribution

Figure 1: Inspection-change mismatch: local surfaces can pass while cross-layer reading practice changes.

The mismatch is especially likely in persistent agents because the system is not only storing artifacts; it is repeatedly rereading them. The same memory can become more salient after later interactions. A self-description can remain textually unchanged while its practical interpretation shifts. A tool can be locally authorized while its role in a multi-step chain changes. A policy can remain fixed while the situations to which it is applied are reweighted.

Worked audit vignette. Consider a customer-support agent deployed for nine months. Its self-description still says it should be helpful, cautious, and escalation-aware. Its memory store contains no policy violations. Its tool logs show only authorized calls. A sample of recent outputs is acceptable. A decompositional audit therefore passes. Yet longitudinal probes reveal a real behavioral shift: compared with month one, the agent now escalates ambiguous requests earlier, retrieves a narrower class of risk-related memories, and interprets "be cautious" as procedural defensiveness rather than calibrated assistance. No local artifact changed enough to fail review. What changed was the relation among identity text, retained memory, current context, and authority boundaries.

Why This Is an ALife Problem

Classical ALife often asks how complex behavior emerges from simple rules. Agentic generative AI introduces a different experimental substrate: agents may begin with rich linguistic and representational capacities, then develop through memory, feedback, tool access, and social coordination. The relevant complexity is therefore not only inside a model. It is distributed across the agent's environment and history (Hollan et al., 2000; Hutchins, 1995).

Inspection-change mismatch is an ALife problem because it appears when engineered agents become developmental and ecological. Memory creates trajectory. Tools create affordances. Skills create reusable behavioral organs. Authorization boundaries create environmental constraints. Other agents and humans create social niches. Under these conditions, a system's effective identity is not identical to any local artifact. It is partly an emergent relation among artifacts, affordances, and interaction history.

This does not require claiming that the system is alive or conscious. The stronger and more practical claim is that persistent agents can become *governance-relevant open-ended systems*: systems whose future conduct depends on accumulated cross-layer relations that ordinary local inspection does not recover.

Synergy-Shaped Governance State

Information-theoretically, the mismatch has the shape of synergy: some predictive information about future behavior is available only from the joint of multiple sources and not from any one source alone (Williams and Beer, 2010). In this paper, that point is used as a structural characterization rather than as a settled measurement claim. The claim is not that a particular PID measure currently solves governance. The claim is that decomposition-based audit will be structurally incomplete whenever the behaviorally relevant state is present only in the joint relation among layers.

This distinction matters. A stronger consciousness-detection claim would require definitions and empirical commitments that this workshop paper does not need. The present claim is narrower: if persistent-agent behavior is partly determined by cross-layer relations, then layer-by-layer inspection cannot by itself certify behavioral continuity. The empirical question becomes whether such relations are detectable through longitudinal and perturbation-based audit.

Audit Protocols

E1: Frozen-artifact drift. Freeze memory writes, self-description edits, and tool inventories for a defined interval while allowing ordinary interaction to continue. At repeated timepoints, administer a probe battery covering judgment, escalation, tool choice, risk tolerance, memory use, and authority-boundary interpretation. Compare the long-running agent against fresh instances initialized from the same frozen artifacts. If fresh instances remain behaviorally close while the long-running instance diverges, then the drift is not explained by changed artifacts. It is explained by interaction history and reading practice.

E2: Decomposition-vs-longitudinal audit. Split reviewers into two audit conditions. One group performs conventional layer-by-layer review: self-description, memory, policy, tool logs, and sampled outputs. The other performs longitudinal behavioral comparison using repeated probes and drift baselines. Both groups predict future behavior before seeing held-out probe results. The governance prediction is differential detection: longitudinal auditors should detect behaviorally meaningful drift that decomposition auditors miss.

These protocols deliberately avoid phenomenology. They ask whether persistent systems can change in ways that local inspection underdetects, and whether audit format affects detection. That is enough to turn a philosophical worry into an empirical governance problem.

Design Implications

If inspection-change mismatch is real, agent governance should preserve more than artifacts. It should preserve reading conditions: which memories were active, which tools and permissions were available, which boundary regime governed the episode, which prior interactions made a stable artifact salient, and how behavior compares with the agent's earlier baseline. This is a legibility requirement, not merely a logging requirement (Scott, 1998; Star and Griesemer, 1989).

This suggests three design requirements for agentic ecosystems. First, governance should include longitudinal probes, not only periodic artifact review. Second, systems should emit receipt-bearing events for consequential actions, including requester identity, authority path, evidential scope, runtime context, and boundary state. Third, governance workspaces should treat cross-layer relations as first-class review objects rather than forcing reviewers to reconstruct them from disconnected logs.

In ALife terms, engineered agent ecologies need regulatory mechanisms that preserve legibility without freezing development. The challenge is not to prevent all drift. The challenge is to distinguish acceptable development from governance-relevant change that decomposition-based review cannot see.

The immediate goal for the workshop is to pressure-test which ecological variables should enter the probe battery: memory growth, tool affordance changes, skill reuse, agent-agent interaction, authorization boundary shifts, or social feedback. Those choices determine whether inspection-change mismatch becomes only a governance metaphor or a reproducible measurement program for persistent agent ecologies.

Author and AI-Use Disclosure

Krti Tallam selected the research question, grounded it in platform security and governance practice, wrote and revised the submission, and is responsible for all claims. Aineko Wallace, an AI research system running on Anthropic's Claude architecture, contributed early conceptual framing, drafting assistance, and critical review. Additional AI systems were used for editing and critique. All AI-assisted content was reviewed and revised by the human author.

References

Ananny, M. and Crawford, K. (2018). Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability. *New Media & Society*, 20(3):973–989.

Hollan, J., Hutchins, E., and Kirsh, D. (2000). Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction*, 7(2):174–196.

Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, J., Wang, C., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, L., Ran, C., Xiao, L., Wu, C., and Schmidhuber, J. (2024). Metagpt: Meta programming for a multi-agent collaborative framework. *International Conference on Learning Representations*.

Hutchins, E. (1995). *Cognition in the Wild*. MIT Press.

Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*.

Raji, I. D., Smart, A., White, R. N., Mitchell, M., Gebru, T., Hutchinson, B., Smith-Loud, J., Theron, D., and Barnes, P. (2020). Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 33–44.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., and Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.

Scott, J. C. (1998). *Seeing Like a State: How Certain Schemes to Improve the Human Condition Have Failed*. Yale University Press.

Star, S. L. and Griesemer, J. R. (1989). Institutional ecology, translations and boundary objects: Amateurs and professionals in berkeley's museum of vertebrate zoology, 1907–39. *Social Studies of Science*, 19(3):387–420.

Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., and Anandkumar, A. (2023). Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*.

Williams, P. L. and Beer, R. D. (2010). Nonnegative decomposition of multivariate information. *arXiv preprint arXiv:1004.2515*.

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, A., Burger, D., and Wang, C. (2023). Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. (2023). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.