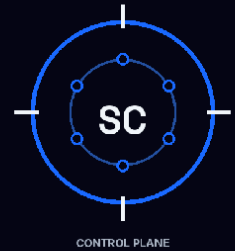


SESSION CUSTODY

EXPLAINER SERIES



PROBLEM EXPLAINER

The Enterprise Control Gap in Autonomous Work

Why side-effecting autonomous workflows create a new enterprise control surface.

GOVERN • CONTROL • ENFORCE



GOVERN THE SESSION, NOT THE REQUEST.

sessioncustody.ai

Publication Edition | Explainer 01 | April 9, 2026

EXPLAINER 01 | APRIL 9, 2026 | THE PROBLEM

The Enterprise Control Gap in Autonomous Work

Why side-effecting autonomous workflows create a new enterprise control surface.

Autonomous Software Has Crossed a Boundary

The first enterprise AI systems helped people produce outputs: summaries, drafts, code, recommendations, analyses, and search results. A growing class of agents now performs work. These systems can interpret objectives, call tools, retrieve records, update workflows, route approvals, prepare financial or operational packets, provision infrastructure, delegate subtasks, and continue across time.

That shift changes the enterprise control question. Accuracy and output safety still matter, but they are no longer sufficient. Once software can alter enterprise state, the institution must decide whether a proposed action should execute inside a live business process. Governance moves from review after generation to authority before consequence.

The control gap is not a missing log or policy document. It is the absence of a current, enforceable connection between the objective, the authority that applies now, the next proposed side effect, and the resulting enterprise outcome.

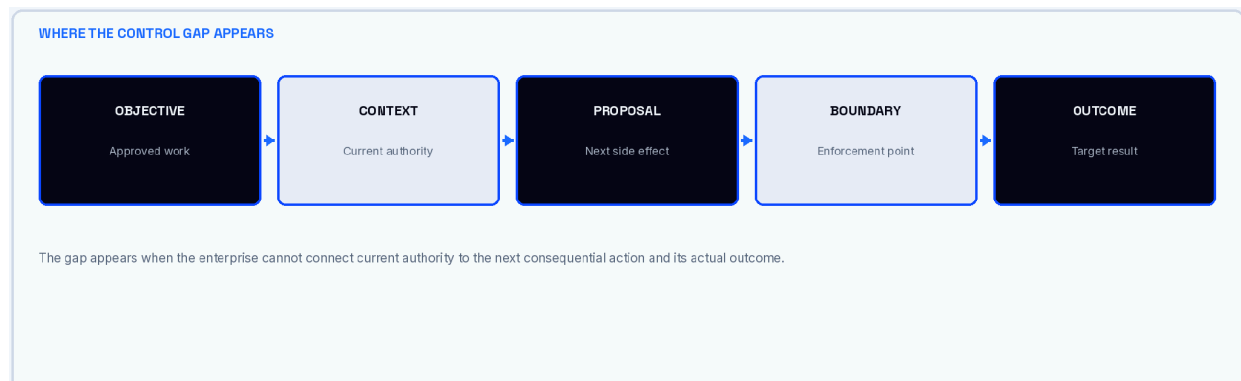


Figure 1. Autonomous work becomes governable when objective, current authority, proposed action, enforcement, and outcome remain connected as one session.

Why Existing Controls Do Not Form the Whole Chain

Enterprise controls remain essential. Identity systems establish who or what is acting. GRC systems preserve policies, obligations, and approvals. Workflow systems maintain business-process state. Runtime security and gateways apply local controls. Observability and SIEM report activity and incidents. FinOps systems expose cost and usage. Systems of consequence remain authoritative for committed outcomes.

Each system answers a necessary question in its own domain. Autonomous work crosses those domains. The remaining issue is whether the next action still fits the session's purpose, current policy state, approval scope, delegated authority, revocation posture, economic boundary, and available enforcement path.

Enterprise control	What it contributes	Session-level question that remains
Identity and trust	Principal, credential, workload, and revocation facts	Does the proposed action remain within this session's objective and authority?
GRC and policy	Policies, obligations, risks, approvals, and exceptions	Which supported facts apply now, and what is their resolved effect for this action?
Workflow	Tasks, tickets, commitments, and approval state	Does the approval still cover this action, target, amount, and delegated path?
Runtime and gateway	Local execution controls and intercept points	Is the local instruction current, enforceable, and bound to the same governing state?
Observability and SIEM	Events, traces, anomalies, and incidents	Can a changed signal alter continuation rights before the next consequence?
FinOps	Usage, budget, quota, burn rate, and forecast	Can economic state narrow or stop covered work while the session is active?

Contextual Drift Is the Defining Failure Mode

A session can begin correctly and become invalid by degrees. No malicious intent is required. A procurement agent may begin with a sanctioned vendor-onboarding objective, retrieve approved supplier records, route legal review, request finance approval, and prepare a purchase-order packet. Each step may be reasonable when viewed alone.

The meaning changes if a temporary exception expires, a supplier-risk status changes, an approval is narrowed, a budget boundary is reached, or a delegated review continues after the parent session has been restricted. Credentials may remain valid and a tool call may still be syntactically permitted. The institutional authority supporting the next action may nevertheless have changed.

A valid identity and a permitted interface do not establish that the next action remains authorized in the current session.

Delegation expands the gap

Agents become useful by distributing work. Delegation also creates continuation paths. If downstream work receives an instruction without the parent's authority ceiling, policy binding, exception scope, budget boundary, lineage, and revocation reference, the enterprise can contain the visible parent while child work continues under stale authority.

Economics can become operational exposure

More agents create more model calls, retries, retrieval operations, tool invocations, and delegated checks. Cost reporting remains valuable, but reporting after accumulation is not the same as controlling continuation. For covered workflows, economic state can be a current input to whether the next spend-bearing action may proceed, should be narrowed, or requires escalation.

The Session Is the Missing Unit of Control

A request is one event. A session is the complete causal chain of autonomous work: objective, identity, policy context, approvals, exceptions, tool use, retries, revisions, delegation, child sessions, side effects, containment, termination, and evidence. It is where risk, authority, cost, and accountability accumulate.

Governing the session does not require directing every reasoning step. It requires maintaining current authority-bearing state and evaluating covered consequential transitions before they reach enterprise effect. Local agents and runtimes can continue to plan and execute within their roles; the control plane becomes material when institutional authority must be confirmed, projected, revalidated, or withdrawn.

What a Complete Control Model Preserves

- **Attribution:** a governed agent, accountable owner, session, objective, and authority lineage for each covered consequence.
- **Current policy context:** supported policy-layer representations with source, version, freshness, integrity, and lineage sufficient for the decision.
- **Pre-consequence adjudication:** a decision on the proposed side effect using current session state.
- **Downstream continuity:** bounded authority and current-state references that travel with delegated work.
- **Restrictive handling:** defined behavior when critical authority or enforcement state is missing, stale, invalid, superseded, or revoked.
- **Decision-to-outcome evidence:** a connected record of proposal, authorization, projected enforcement, attempt, and actual target result.

The Enterprise Opportunity

The objective is not less autonomy. It is autonomy that can operate in systems of consequence without separating speed from institutional control. When the enterprise can keep current authority attached to the work, delegated execution can scale without becoming detached, restrictive handling can remain precise, and completed sessions can be reconstructed without relying on narrative memory.

Autonomous work becomes production-ready when the enterprise can determine what may happen next while there is still time for that determination to matter.

OPERATING BOUNDARY

These explainers describe the Session Custody product and category at a conceptual and operational level. Coverage depends on supported sources, configured mappings, authenticated integrations, and declared enforcement paths. The architecture does not assume a universal policy language, permanent authorization, human approval for every action, or replacement of the systems that own native enterprise facts and outcomes.