

BUSINESS EXPLAINER

Cognous Agent Replay Bundle

Portable replay records for AI-agent runs

A log records events. A replay bundle reconstructs the agent run.

Business explainer prepared from the Cognous Agent Replay Bundle whitepaper (de Lima, July 2026).

1. Executive Summary

Cognous Agent Replay Bundle is an open-source reference format for packaging AI-agent run evidence into portable replay records. It helps organizations reconstruct an agent run after the fact by preserving, in one artifact, the run context, the actions the agent proposed, the policy decisions made in response, the rule-level evaluation traces behind those decisions, the actions that were blocked, the authority that existed, the sources the agent relied on, the final output, and the validation, redaction, and signature metadata that make the record reviewable and shareable.

It is deliberately limited in what it claims to be. It does not run the agent. It does not enforce policy. It does not certify compliance. It does not prove that the agent was correct. It does not replace security, authorization, audit, legal, or governance processes. What it does is create the structured technical evidence artifact those processes can inspect — the run-level package that makes reconstruction possible.

The central message for enterprise leaders is this:

As enterprises move AI agents from demos into production, logs are not enough. Teams need portable replay records that show what the agent proposed, what policy decided, what was blocked, what authority existed, what sources were relied upon, and what evidence can be reviewed later.

The distinction is simple. A log records events — scattered, low-level, organized around systems. A replay bundle reconstructs the agent run — coherent, complete, organized around the question a reviewer actually asks: what happened here, and can we see it?

2. The Business Problem

AI agents are beginning to:

- call tools and APIs;
- access enterprise data;
- generate recommendations;
- propose actions;
- draft messages;
- trigger workflows;
- rely on documents, databases, APIs, and user input;
- produce outputs that influence business decisions.

This creates a replay gap. Runtime systems may produce logs, traces, tool-call records, policy decisions, validation outputs, redaction metadata, or signatures — but these artifacts are often scattered across systems, inconsistent in format, too technical to review, and, above all, not organized around the agent run. When a question arises about a specific run, someone has to reassemble the story by hand.

That gap shows up as seven recurring pain points.

1. Logs are not replay packages. A log line may show that a function was called. It does not preserve the agent-level story: what was proposed, what was decided, what was blocked, what authority existed, and what was relied on — in one coherent, portable package.

2. Action proposals are often lost. For governed agents, the proposal matters. An agent may propose an action that is never executed because policy blocked or escalated it. If only executed actions are recorded, teams lose the evidence of what the agent attempted.

3. Blocked actions are not visible enough. Blocked actions are governance evidence, not failures to hide. A blocked outbound email or blocked purchase order demonstrates that controls operated as intended — but only if the block is recorded and preserved.

4. Policy decisions need context. A decision is not just an outcome. Reviewers need to know which action was evaluated, which policy applied, what result was returned, why, and when — with the decision linked to the proposal it governed.

5. Policy evaluation needs traceability. An allow, block, or escalate result is easier to review when the rule-level evaluation path behind it is preserved: which rules were checked, which matched, and why the final result followed.

6. Authority and reliance are often reconstructed too late. After an incident, audit, or customer dispute, teams need to know what authority existed and what sources the agent relied

on. If those records were not captured at or near the run, reconstruction becomes costly and unreliable.

7. Exports need redaction and integrity metadata. Run records may contain sensitive payloads, targets, and outputs. Sharing them with auditors, counsel, customers, or regulators requires redaction — and a way to detect whether the exported record was modified afterward.

These pain points are concrete. Consider four common deployments:

- A **customer-service agent** whose proposed outbound email was blocked — a proposal and a block that both need to survive as evidence.
- A **procurement agent** whose purchase-order proposal was blocked pending authority.
- A **finance workflow agent** operating under payment approval controls whose runs must be reviewable.
- An **internal research agent** whose exports are subject to review and whose summaries depend on document sources.

In each case, the run happened quickly, across several systems, under specific policy conditions. The business question is whether, weeks or months later, anyone can put the run back together.

3. What Agent Replay Bundle Is

Agent Replay Bundle is a portable technical evidence package for AI-agent runs. It preserves, in a single validated artifact, the records needed to reconstruct what happened during a run — from the task the agent was given through the proposals, decisions, blocks, authority, reliance, and output that followed.

In one sentence:

It gives enterprises a structured answer to the question: “Can we reconstruct this agent run and review what happened?”

The phrase “replay bundle” is worth translating into business terms. An agent replay bundle is:

- a **run-level evidence artifact** — one package per run, organized around the run;
- a **portable package of technical records** — it travels across systems, teams, and reviews;
- a **reconstruction aid** — for audit, incident response, and deployment governance;
- a **source of supporting evidence** — the technical layer beneath business-facing evidence packs;

- **not a runtime system** — it is produced by or around one;
- **not a policy enforcement system** — it records decisions, it does not make them;
- **not a compliance certificate** — it preserves evidence, it does not certify obligations.

In the Cognous Open Control Stack — Declare → Control → Replay → Evidence — Agent Replay Bundle is the Replay layer: the portable record between runtime control and business-facing review. Section 9 describes the full stack.

4. What It Does

Preserves the run frame

The bundle records the context in which the run occurred: the task, the actor, the environment, the tools that were allowed and blocked, the active policy version, and the timestamp.

Business value: reviewers understand the conditions under which the agent operated. Without the frame, individual proposals and decisions are difficult to interpret.

Preserves action proposals

The bundle records what the agent proposed — the tool, action type, target, payload, timestamp, and reason — whether or not the action was ever executed.

Business value: teams can inspect attempted behavior, not only completed behavior. For governed agents, the proposal is often the most important record.

Preserves policy decisions

The bundle records whether each proposed action was allowed, blocked, or escalated, including the policy name, reason, timestamp, a deterministic fingerprint, and a reference to the evaluation trace where one exists.

Business value: reviewers can see how runtime governance responded to each agent proposal, decision by decision.

Preserves policy evaluation traces

The bundle can record rule-level evaluation detail: which rules were checked, which matched, what each returned, and how the final result followed.

Business value: teams can inspect why a policy decision occurred, not just what it was.

Records blocked actions

The bundle treats blocked actions as first-class records, each linked to the proposal it stopped and the block decision that stopped it.

Business value: blocked actions become evidence that governance controls operated — preserved, not lost as non-events.

Records authority context

The bundle records the authority relevant to the run: actor, scope, source, validity, and expiration. Authority records preserve asserted authority context; they do not grant authority.

Business value: reviewers can evaluate whether actions were associated with bounded, current authority at the time of the run.

Records reliance sources

The bundle records the sources the agent relied on — tools, databases, files, APIs, user input, or model outputs — with links to the actions they supported where possible.

Business value: teams can evaluate provenance and source dependence after the run: what stood behind the output.

Preserves final output

The bundle can include the final output produced by the run, where appropriate and permitted.

Business value: reviewers can connect proposals, policy decisions, reliance, and output in one package rather than across systems.

Validates internal consistency

The reference implementation checks cross-references and consistency: every decision must reference a known proposal, every blocked action must correspond to a block decision, and trace fingerprints must match their linked decisions.

Business value: teams can identify malformed or incomplete run records before relying on them for review.

Supports redaction

A bundle can be redacted for review or export: sensitive payloads, targets, outputs, and reasons can be removed while identifiers, timestamps, decision results, fingerprints, and cross-references are preserved, and the redacted fields are recorded in redaction metadata.

Business value: sensitive content can be removed while the governance structure of the run remains fully inspectable.

Supports export integrity

The reference implementation supports signing exported bundles so that reviewers can detect whether a bundle changed after it was signed. This is export-integrity metadata — useful and inspectable — but it is not production key management and not non-repudiation.

Business value: exported evidence carries a tamper-evidence signal, with the limits of that signal stated plainly.

5. Why It Is Valuable

Better incident reconstruction. When a question arises about a run, teams can inspect what the agent proposed, what policy decided, what was blocked, what was relied on, and what output was produced — from one artifact instead of a forensic hunt.

Better audit preparation. Audit teams review structured run evidence instead of piecing together logs, traces, and notes. The bundle is the run-level record that audits keep asking for and rarely find.

Better governance evidence. Replay bundles supply the technical records beneath business-facing evidence packs, so deployment reviews can trace summaries back to runs.

Better blocked-action visibility. Blocked actions are preserved as proof that controls operated — the difference between “nothing bad happened” and “something inappropriate was attempted and stopped, and here is the record.”

Better authority and reliance review. The bundle records who or what had authority during the run and which sources the agent used, so those questions can be answered from evidence rather than memory.

Better export discipline. Redaction and signing support safer sharing of run evidence with auditors, counsel, customers, or regulators — structure preserved, sensitive content removed, modification detectable.

Better cross-functional review. Engineering, security, risk, compliance, legal, and audit teams reference the same run artifact instead of maintaining parallel partial reconstructions.

6. How It Is Different from Other Approaches

Enterprises already produce several kinds of records that touch agent behavior. Agent Replay Bundle overlaps with none of them exactly, and the differences matter.

Versus logs

Logs record events: service calls, errors, low-level activity. They are indispensable, but they do not preserve the complete agent-level reconstruction package — proposals, decisions, blocks, authority, and reliance in one coherent record.

Difference: Logs show activity. Agent Replay Bundle organizes run evidence.

Versus traces

Distributed traces show execution paths through systems. They are built for performance and debugging, and they may not preserve governance concepts at all: action proposals, policy decisions, blocked actions, authority records, or reliance records.

Difference: Traces show system behavior. Replay bundles package governance-relevant run evidence.

Versus dashboards

Dashboards visualize data. They are views, and they are only as good as the records beneath them.

Difference: A dashboard is a view. A replay bundle is a portable evidence artifact that a dashboard could display.

Versus governance reports

Governance reports summarize posture for business audiences. They are the output of review, not the evidence beneath it.

Difference: Reports are business-facing summaries. Replay bundles are the technical run evidence that can support those summaries.

Versus Agent Governance Evidence Pack

Agent Governance Evidence Pack — the Evidence layer of the same stack — translates runtime records into business-facing governance evidence for deployment review. It inventories and references replay bundles rather than replacing them.

Difference: Replay bundles answer “what happened in this run?” Evidence packs answer “what evidence supports this deployment review?”

Versus ODES

ODES, the Open Decision Evidence Standard, is a separate proposed open, vendor-neutral public standard for portable decision evidence in AI-mediated and cross-boundary decisions. Agent Replay Bundle is not ODES and does not define ODES conformance; Section 10 describes the boundary.

Difference: ODES focuses on portable decision evidence for relying parties across boundaries. Agent Replay Bundle focuses on reconstructable technical run evidence for AI-agent runs.

Comparison at a glance

Category	What it does	What it misses	How Agent Replay Bundle differs
Logs	Record low-level events	The coherent agent-level run story	Packages proposals, decisions, blocks, authority, and reliance around the run
Traces	Show execution paths through systems	Governance concepts: proposals, decisions, blocks, authority, reliance	Preserves governance-relevant run evidence, not just execution flow
Dashboards	Visualize data	A portable, validatable evidence artifact	Is the run artifact a dashboard could display
Governance reports	Summarize posture for business audiences	The technical evidence beneath the summary	Supplies the run-level records that support the report
Agent Governance Evidence Pack	Translates records into business-facing deployment evidence	Run-level technical reconstruction detail	Answers the run question; the pack answers the deployment question
ODES records	Proposed standard for portable cross-boundary decision evidence	AI-agent run reconstruction as such	Native replay artifact for agent runs; may supply supporting evidence to ODES-style records

Table 1. Agent Replay Bundle compared with adjacent records and systems.

7. What It Is Not

Clear scope is part of the design. Cognous Agent Replay Bundle is **not**:

- an agent framework;
- a model runtime;
- a planner;
- a runtime control plane;
- a policy engine;
- a replay viewer;
- a hosted service;
- a compliance certification product;
- a complete enterprise governance platform;

- a security boundary by itself;
- a substitute for application authorization;
- a guarantee that model outputs are correct;
- a guarantee that regulatory obligations have been satisfied.

Its value is focused: it packages run evidence for reconstruction and review. The quality of a replay record depends on the upstream systems and governance processes that produce it. Authority records preserve asserted context but do not grant authority; reliance records document sources but do not prove those sources were correct; traces record the rule path but do not prove policy sufficiency; and signing detects modification but does not, by itself, establish trust. The bundle makes the run inspectable — it does not make the run right.

8. Example Scenario

Consider a customer-service agent deployed in a controlled environment, asked to help respond to a customer issue.

1. **The run frame records the context:** the task, the acting agent, the environment, the allowed and blocked tools, and the active policy version.
2. **The agent proposes reading CRM records.** The proposal is preserved with tool, action type, and target.
3. **The policy system allows the CRM read,** and the decision — with its reason, fingerprint, and trace reference — is recorded.
4. **The agent relies on CRM and account-note sources,** and each reliance is recorded and linked to the actions it supported.
5. **The agent drafts an outbound email** responding to the customer's issue.
6. **The agent proposes sending the email externally.** The proposal is preserved exactly as made.
7. **Policy blocks the outbound send** because external-send authority is missing for this run.
8. **The blocked action is recorded as first-class evidence,** linked to both the proposal and the block decision.
9. **The final output notes** that the draft requires review or escalation before any send.
10. **The bundle validates:** every decision references a known proposal, the blocked action corresponds to a block decision, and trace fingerprints match their decisions.
11. **Sensitive payloads are redacted** before broader review — customer content removed, structure and decision results preserved, redacted fields recorded.
12. **The redacted export is signed** so later reviewers can detect modification after export.

13. **The replay bundle is referenced** by a business-facing governance evidence pack supporting the deployment review.

The business outcome: the review team can reconstruct what happened, see that the unauthorized send was proposed and blocked, inspect what sources the draft relied on, and preserve the run evidence — redacted and signed — for audit or incident review. The run is no longer a story someone has to retell; it is a record anyone can open.

9. Relationship to the Cognous Open Control Stack

Agent Replay Bundle is one component of a four-layer pattern for governed agent deployment:

Declare → Control → Replay → Evidence

- **Agent Action Manifest** declares what actions an agent may propose, before runtime.
- **Agent Control Plane** records and governs runtime proposals and decisions.
- **Agent Replay Bundle** packages technical run records into portable replay artifacts.
- **Agent Governance Evidence Pack** translates technical records into business-facing governance evidence.

The stack separates operational concerns that are related but distinct. Declaration establishes the expected action surface; runtime control governs actual behavior; replay makes each run portable and reconstructable; evidence makes the whole reviewable by business stakeholders. A replay bundle can be produced by a control plane, assembled from other runtime records, or constructed by any system that implements the public format — the layer is defined by the artifact, not by a single producer.

10. Relationship to ODES

ODES, the Open Decision Evidence Standard, is a separate proposed open, vendor-neutral public standard for portable decision evidence in AI-mediated and cross-boundary decisions. Agent Replay Bundle is not ODES, does not define ODES, and should not be described as ODES-conformant unless a future ODES profile and conformance test suite define what that claim means.

The relationship is optional and supporting, not identity. A replay bundle may provide technical run evidence that could support, or be referenced by, an ODES-style portable decision evidence record — particularly where an AI-agent action or decision crosses an organizational, system, or jurisdictional boundary. In such a case, the replay bundle contains evidence about the action proposal, the policy decision, the authority context, the reliance sources, and the validation,

redaction, and signature metadata. A future optional mapping could translate selected replay-bundle fields into ODES-compatible records.

The distinction to keep straight:

- **ODES** — a proposed public standard for portable decision evidence.
- **Agent Replay Bundle** — a Cognous reference format for portable technical replay records of AI-agent runs.
- **The relationship** — optional mapping and supporting evidence, not identity and not conformance.

11. Open-Source Reference Implementation

Agent Replay Bundle is published as a public, open-source reference implementation. The repository is public so teams can inspect the format, run the examples, validate bundles, redact records, sign exports, verify signatures, and build compatible replay workflows before making any platform commitments.

In practical terms, the repository includes:

- published JSON schemas defining the replay-bundle format, so other systems can produce and consume bundles;
- a Python package with a clearly bounded public object model (built on Pydantic);
- a semantic validator that checks cross-references and consistency beyond basic structure;
- a redaction helper that removes sensitive content while preserving governance structure;
- a signing helper for export-integrity metadata, with its limitations documented;
- a command-line interface for validating, summarizing, redacting, signing, and verifying bundles;
- example bundles for common scenarios — customer service, internal research, procurement, plus redacted and signed examples;
- a test suite run through GitHub Actions across supported Python versions;
- an Apache-2.0 license.

The implementation is intentionally small — a format, a validator, and export helpers, meant to be readable and verifiable by an enterprise platform team, not a platform in itself.

12. Strategic Summary

Agent Replay Bundle gives enterprises a practical way to preserve AI-agent run evidence in a portable, reviewable form. It creates structure where evidence is today scattered across logs,

traces, tool calls, policy systems, redaction workflows, and manual review notes. It helps engineering, security, risk, compliance, legal, audit, and business stakeholders reconstruct agent behavior using a shared technical artifact — one that validates, redacts, and signs without losing the governance structure of the run.

It does not claim to enforce policy, certify compliance, or prove correctness. It claims something narrower and prior to all of those: that a run must be reconstructable before it can be meaningfully reviewed, and that reconstruction should rest on a portable record rather than on whatever fragments survive.

Before enterprises can scale governed AI agents, they need more than logs. They need portable replay records. That is the role of Agent Replay Bundle.