

BUSINESS EXPLAINER

Cognous Replay Bundle

Portable reconstruction records for AI-agent runs

October 2026

A log records events. A replay bundle reconstructs what the retained records support.

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

1. Executive Summary

Cognous Replay Bundle (formerly Cognous Agent Replay Bundle) is an open-source reference format, importer, validator and command-line tool for packaging AI-agent run evidence into portable reconstruction records. It helps organizations put an agent run back together after the fact: what was proposed, what was decided, what was attempted, what was observed at the destination, and which system supplied each of those facts.

Since the first release in July 2026 the component has grown in one important direction. It still offers the original replay-bundle format for packaging a run frame, proposals, policy decisions, traces, blocked actions, authority and reliance records. It now also offers **Reconstruction Bundle 0.2.0**, which imports the actual records retained by specific, pinned versions of the Cognous Control Plane and Execution Runtime. Those records include uncertain outcomes, such as a timeout where nobody knows yet whether the action took effect.

It is deliberately limited in what it claims to be. It does not run the agent. It does not enforce policy or re-evaluate it. It does not renew authorization, repeat an action or verify delivery independently. It does not certify compliance. What it does is create the structured technical evidence artifact those processes can inspect, with gaps and uncertainty left visible rather than smoothed over.

The central message for enterprise leaders is this:

As AI agents begin to take consequential actions, logs are not enough. Reviewers need a connected, portable record of proposals, decisions, attempts, observations and their sources. That record has to admit what is unknown instead of telling a retrospective success story.

The distinction is simple. A log records events: scattered, low-level and organized around systems. A replay bundle reconstructs the agent run. It is organized around the question a reviewer actually asks: what happened here, what do the retained records support, and what remains unresolved?

2. The Business Problem

AI agents are beginning to:

- call tools and APIs;
- access enterprise data;
- propose actions such as refunds, messages, purchases or exports;
- trigger workflows that change records in other systems;
- rely on documents, databases, APIs and user input;
- produce outputs that influence business decisions.

This creates a replay gap. Authority checks, execution and recovery each leave their own records: logs, decisions, attempt rows, destination observations, redaction notes and signatures. These records are often scattered across systems and inconsistent in format. They can disagree with each other or leave out important facts. When a question arises about a specific run, someone has to reassemble the story by hand.

That gap shows up as eight 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 in one coherent, portable package: what was proposed, what was decided, what was attempted and what was observed.
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 held 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 refused refund or blocked outbound email shows that a control operated, but only if the refusal is recorded and preserved.
4. **Policy decisions need context.** Reviewers need to know which proposal a decision applied to, which policy and grant were in force, what result was returned, why, and when.
5. **Policy evaluation needs traceability.** An allow, block or escalate result is easier to review when the rule-level path behind it is preserved.
6. **Authority and reliance are often reconstructed too late.** After an incident, audit or dispute, teams need to know what authority was in force and what sources the agent relied on. Records captured long after the run are costly and unreliable.
7. **Outcomes can be uncertain.** A request can time out before the destination acknowledges it. The action may or may not have taken effect. A good record keeps that "unknown" visible. It does not turn a missing acknowledgement into a failure, a success or permission to try again.
8. **Exports need redaction and integrity metadata.** Run records may contain sensitive payloads and outputs. Sharing them with auditors, counsel or customers requires redaction, plus a way to detect whether the exported record was changed afterward.

These pain points are concrete. The reference workflow in the Cognous Open Control Stack is a single **synthetic refund**. A refund proposal is checked against independently supplied authority, executed against a synthetic local database, and sometimes left with an unresolved delivery status. The business

question is whether, weeks or months later, anyone can put that run back together accurately, including the parts that were never resolved.

3. What Replay Bundle Is

Replay Bundle is a portable technical evidence package for AI-agent runs. It keeps, in one validated artifact, the records needed to reconstruct a run, and it records where each record came from.

In one sentence:

It gives enterprises a structured answer to the question: "What do the retained records of this agent run actually support, and what remains unknown?"

The phrase "replay bundle" is worth translating into business terms. A replay bundle is:

- **a run-level evidence artifact:** one package organized around the run;
- **a portable package of technical records** that travels across systems, teams and reviews;
- **a reconstruction aid** for audit, incident response and deployment review;
- **a source of supporting evidence** beneath business-facing evidence packs and portable decision-evidence records;
- **non-effecting:** importing records never re-runs the agent, re-evaluates policy, renews authority or repeats an action;
- **not a runtime system:** it is produced from records that other systems retain;
- **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.

The word "replay" refers to reconstructing the record. It does not mean re-executing the run. The Open Control Stack hub also runs an experiment it calls "paired replay", which compares the same frozen request under different enforcement conditions. That experiment is unrelated to Replay Bundle reconstruction.

In the Cognous Open Control Stack, Replay Bundle sits in responsibility group **05 · Replay & Evidence**, alongside the Governance Evidence Pack and ODES. Section 9 describes the full stack.

4. What It Does

Packages run records in the original replay format

The original replay-bundle format (format version "0.1") is still available and unchanged. It records the run frame (task, actor, environment, allowed and blocked tools, policy version), action proposals, allow/block/escalate policy decisions with deterministic fingerprints, rule-level evaluation traces, blocked actions, authority records, reliance records and the final output.

Business value: any system can describe an agent run in a simple, framework-neutral structure that reviewers can read.

Imports actual producer records

Reconstruction Bundle 0.2.0 imports the records that specific Control Plane and Execution Runtime revisions actually retain: proposals, decisions, decision bindings, effects, attempts, observations and reconciliations. Each supported combination is pinned to an exact source revision. An unknown revision is rejected. The importer never guesses compatibility because two records look alike.

Business value: the reconstruction is built from what the producing systems recorded, not from a hand-assembled summary, and it cannot quietly misread a record format it does not know.

Preserves identity and lineage

The importer keeps the identity of each proposal, decision and effect. The Control Plane and the executor each number their execution attempts. Replay keeps those two sets of attempt numbers apart, so that matching labels are never treated as proof that two records describe the same attempt. Records are linked only where the producer evidence actually supplies the link.

Business value: reviewers can follow one refund from proposal to destination without the record inventing connections that the systems never made.

Retains uncertain and rejected outcomes

An unknown acknowledgement stays unknown. The record also keeps observations that were rejected as stale, malformed or tied to the wrong effect. Missing observations stay missing, and an observed absence is never treated as permission to retry. Recovery attempts keep their own history and lineage.

Business value: the record shows what was uncertain at the time instead of a tidied-up story told afterward.

Reports gaps explicitly

Each import produces a machine-readable report. It lists how fields were mapped, which fields went unmapped, which dependencies were missing and which semantics are unsupported. Values can be marked absent, unknown, unavailable or redacted. A bundle is labelled "reconstruction complete" only when the declared import inputs were fully represented. That label says nothing about whether the refund itself was completed.

Business value: reviewers can tell the difference between "the record is complete" and "the action succeeded," which are separate questions.

Checks consistency

The original format's validator checks cross-references: every decision must reference a known proposal, every blocked action must match a block decision, and trace fingerprints must match their decisions. The importer goes further for producer records. It checks that the proposal, decision binding, execution envelope and destination row describe the same operation, and it rejects contradictory content recorded under the same decision, attempt or effect identity.

Business value: malformed, incomplete or contradictory run records are caught before anyone relies on them.

Supports redaction

Both formats can be redacted. In the original format, sensitive payloads, targets, outputs and reasons can be removed while identifiers, timestamps, decision results, fingerprints and cross-references are kept. In Reconstruction Bundle 0.2.0, a redacted copy receives a new identity linked to its source. The original integrity data is discarded, and affected source digests are downgraded to attributed claims. An old signature is never presented as covering modified content.

Business value: sensitive content can be removed while the governance structure of the run stays inspectable and honestly labelled.

Supports export integrity

Bundles can be signed with HMAC-SHA256 using a shared secret, so a holder of that secret can detect whether an export changed after signing. This is export-integrity metadata. It is not production key management, not a publicly verifiable issuer signature and not non-repudiation. A caller-supplied key identifier is only a label.

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 was proposed, what was decided, what was attempted and what the destination showed, all from one artifact instead of a forensic hunt.

Honest handling of uncertainty. A timed-out refund remains visibly unresolved until an accepted observation resolves it. Reviewers can see what was known at each point without the record claiming more.

Better audit preparation. Audit teams review structured run evidence with explicit source attribution instead of piecing together logs, traces and notes.

Better governance evidence. Reconstruction bundles supply the technical records beneath the Governance Evidence Pack and ODES exports, so deployment reviews can trace a summary back to the run that produced it.

Better blocked-action visibility. A refused or blocked action is preserved as evidence that a control operated. That is the difference between "nothing bad happened" and "something inappropriate was attempted and stopped, and here is the record."

Better export discipline. Redaction and signing support safer sharing of run evidence, with redacted derivatives clearly distinguished from originals.

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

These are the purposes the format is designed for. The reference tests have not measured review time, cost or enterprise outcomes, and no such benefit is claimed.

6. How It Is Different from Other Approaches

Enterprises already produce several kinds of records that touch agent behavior. Replay Bundle is not the same as any of them, and the differences matter.

Versus logs

Logs record events: service calls, errors and low-level activity. They are indispensable, but they do not preserve proposals, decisions, attempts, observations and their sources in one coherent record.

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

Versus traces

Distributed traces show execution paths through systems. They are built for performance and debugging and may not preserve governance concepts at all.

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 them.

Versus Governance Evidence Pack

Cognous Governance Evidence Pack, in the same responsibility group, prepares business-facing review material. Its traceable importer combines the Action Manifest with Reconstruction Bundle input and records each transformation, so a reviewer can follow a summary back to its source. It builds on reconstruction bundles rather than replacing them.

Difference: Replay bundles answer "what do the run's records support?" Evidence packs answer "what evidence supports this deployment review?"

Versus ODES

ODES, the Open Decision Evidence Standard, is a separate open, vendor-neutral discussion draft with reference tooling for portable decision evidence. Replay Bundle is not ODES and does not define ODES conformance. Section 10 describes the boundary.

Difference: ODES carries decision evidence for a recipient to evaluate under its own policy. Replay Bundle reconstructs technical run evidence for AI-agent runs.

Comparison at a glance

Category	What it does	What it misses	How Replay Bundle differs
Logs	Record low-level events	The coherent agent-level run story	Packages proposals, decisions, attempts and observations around the run
Traces	Show execution paths through systems	Governance concepts such as proposals, decisions, authority and observations	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
Governance Evidence Pack	Prepares traceable business-facing review material	Run-level reconstruction detail	Answers the run question; the pack answers the deployment question
ODES records	Carry portable decision evidence for a recipient's evaluation	AI-agent run reconstruction as such	Native reconstruction artifact; ODES tooling can export from it

7. What It Is Not

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

- an agent framework;
- a model runtime;
- a planner;
- a runtime control plane;
- a policy engine, or a re-evaluation of policy;
- an authorization system: it never renews or grants permission;
- a delivery verifier: it does not query the destination or confirm settlement;
- a replay viewer;
- a hosted service;
- a compliance certification product;
- a complete enterprise governance platform;
- a security boundary by itself;
- 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. A replay record is only as good as the 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. A complete reconstruction is not proof that the action finished. Signing detects

modification but does not, by itself, establish trust. No signature, digest, message receipt or evidence package authorizes execution. The bundle makes the run inspectable. It does not make the run right.

8. Example Scenario

Consider the stack's reference workflow: a synthetic refund processed by an AI agent in a controlled test environment.

1. A refund proposal is declared against an Action Manifest and submitted to the Control Plane.
2. A separately configured resolver supplies the current authority context. The Control Plane decides, then re-checks the decision-critical inputs immediately before dispatch.
3. The Execution Runtime attempts one identified effect, E1, against a synthetic SQLite destination.
4. The caller times out before acknowledgement. Delivery status is **unknown**, not failed.
5. A later reconciliation observes the destination for the original effect E1. The timeout is not treated as permission to send a replacement.
6. The importer receives the retained Control Plane record, the original proposal and the executor export, and checks that the combination of source revisions is supported.
7. The reconstruction keeps the decision, the binding, both attempt namespaces, the unknown acknowledgement and the later observation as separate records, each attributed to its source.
8. The import report marks any missing dependency or unmapped field. The observation history for E1 shows the latest accepted observation without rewriting the earlier unknown acknowledgement.
9. Import changes nothing: no policy is re-evaluated, no authority is renewed and no destination is touched.
10. A redacted derivative is produced for broader review. It receives its own identity, linked to the source bundle.
11. The derivative is signed with a shared-secret HMAC so later reviewers holding the secret can detect modification.
12. The Governance Evidence Pack importer and the ODES exporter can derive review material from the retained reconstruction without rerunning the action.

The business outcome: the review team can see that the refund was authorized, attempted and left briefly unresolved, and how reconciliation resolved it, without the record ever claiming more than its sources support. The run is no longer a story someone has to retell. It is a record anyone can open.

The original customer-service example is still in the repository as a legacy replay bundle. In that example, an agent's outbound email proposal is blocked for lack of external-send authority, and the bundle preserves both the proposal and the block.

9. Relationship to the Cognous Open Control Stack

The original design described a four-layer sequence: Declare, Control, Replay, Evidence. The current Cognous Open Control Stack has **13 components in six responsibility groups**, plus a cross-cutting exchange layer. The groups are a map of responsibilities. They are not an execution sequence, and no one needs to install every component.

Group	Components
01 · Knowledge & Provenance	Evidence Registry (The Index), Evidence Attestation (BitRep)
02 · Institutional Governance	Institutional Governance (Alvorada)
03 · Declaration & Policy Control	Action Manifest, Control Plane
04 · Constrained Execution	Execution Runtime (Moltbot Safe)
05 · Replay & Evidence	Replay Bundle, Governance Evidence Pack, ODES
06 · Reasoning & Human Oversight	PRP, Research Intelligence, TFA

Governed Exchange (GAX/IMX) connects governed messages, recipient assessment and continuity across the groups.

Replay Bundle's neighbours in the supported workflow are:

- **Action Manifest**, which declares what an agent may propose. Manifest 1.1 adds runtime bindings that let a controller compare a proposal with a specific declared action.
- **Control Plane**, which evaluates proposals against independently resolved authority, revalidates decision-critical inputs at effect time, and keeps persistent decision, attempt and reconciliation records.
- **Execution Runtime**, which executes bounded operations against the synthetic destination and exports its attempts and observations.
- **Governance Evidence Pack**, which turns the Manifest and reconstruction input into traceable business-facing review material.

A replay bundle can still be produced by any system that implements the public format. The 0.2.0 producer importers, however, accept only the exact producer revisions they were qualified against.

10. Relationship to ODES

ODES, the Open Decision Evidence Standard, is a separate open, vendor-neutral discussion draft with reference tooling for portable decision evidence in AI-mediated and cross-boundary decisions. Replay Bundle is not ODES and does not define ODES. It should not be described as ODES-conformant.

The relationship is optional and supporting; the two are not the same thing. Since July, the ODES repository has added its own documented field mapping from exact supported Action Manifest and Replay inputs into ODES packages. That mapping is owned and versioned in ODES, not in Replay Bundle. Exporting a package does not make the reconstruction an ODES record, and it does not settle whether a recipient should rely on the underlying decision.

The distinction to keep straight:

- **ODES:** a discussion-draft standard for portable decision evidence, evaluated by each recipient under its own policy.
- **Replay Bundle:** a Cognous reference format for portable reconstruction records of AI-agent runs.
- **The relationship:** an optional mapping and supporting evidence. It is neither identity nor conformance.

11. Open-Source Reference Implementation

Replay Bundle is published as a public, open-source reference implementation at github.com/cognous/cognous-replay-bundle (renamed from `cognous-agent-replay-bundle` in October 2026). The package name, `agent-replay-bundle`, and the `arb` command did not change.

In practical terms, the repository includes:

- published JSON schemas for the original replay format and for Reconstruction Bundle 0.2.0;
- a Python package (version 0.2.0, Python 3.11 or later, built on Pydantic);
- revision-pinned importers for supported Control Plane and Execution Runtime producer records;
- a semantic validator, a redaction helper and an HMAC export-integrity helper for each format;
- a command-line interface for validating, summarizing, redacting, signing and verifying original-format bundles;
- example bundles: customer service, internal research, procurement, redacted and signed, plus generated bounded-refund reconstructions and producer examples;
- an optional, separate recovery-inspection module for synthetic recovery evidence, which is not part of the hub's selected release;
- test suites run through GitHub Actions on Python 3.11 and 3.12, plus separate jobs that check out the exact producer revisions;
- an Apache-2.0 license.

A first look requires only a few commands:

```
python -m pip install -e ".[dev]"
arb validate examples/customer_service_replay_bundle.json
arb summarize examples/customer_service_replay_bundle.json
```

The full selected integration is run from the hub quickstart's clean-checkout verification for the current selected lock (PV-3), which supplies the exact producer checkouts.

What the evidence supports today. The hub selects this component with the Reconstruction Bundle 0.2.0 interface, as part of a merged, pinned bounded synthetic reference (lock profile merged-producers-v1). It uses a synthetic SQLite destination on one host. In the current release evidence (CI run 37694032916), all 35 acceptance scenarios were resolved and the required suites were executed

twice. A fresh 13 October reproduction ran 1,009 Python testcases in each of two repetitions plus 20 Node tests, with 34 invariants passing and one characterized limitation. These are stack-wide counts, not a Replay Bundle count. An earlier 915-test figure is historical.

What it does not establish. This is not a production deployment, a compliance certification or proof of institutional adoption. The default control path re-checks authority before an effect but is not atomic with the destination commit. An optional same-host transaction profile exists but is not enabled by default, and no distributed guarantee is claimed. Deduplicating by effect identifier is not the same as deduplicating equivalent business requests. Feature intake was frozen on 13 October. Public preview approval is pending, and final publication and operational trust remain on hold.

12. Strategic Summary

Replay Bundle gives enterprises a practical way to keep AI-agent run evidence in a portable, reviewable form. It creates structure where evidence is otherwise scattered across logs, decision records, execution attempts, destination observations, redaction workflows and manual notes. It lets engineering, security, risk, compliance, legal, audit and business stakeholders reconstruct agent behavior from one shared technical artifact. That artifact keeps track of where each fact came from and admits what is still unknown.

It does not claim to enforce policy, re-authorize actions, verify delivery, certify compliance or prove correctness. It claims something narrower that comes before all of those: a run must be reconstructable before it can be meaningfully reviewed, and the reconstruction should rest on the records the systems actually retained rather than on whatever fragments survive.

Before enterprises can scale governed AI agents, they need more than logs. They need portable reconstruction records that are honest about uncertainty. That is the role of Cognous Replay Bundle.