

BUSINESS EXPLAINER

Cognous Governance Evidence Pack

Turn traceable runtime records into reviewable governance evidence

October 2026

A replay bundle can reconstruct a run. An evidence pack helps a business reviewer understand the deployment, without pretending to be more than the records behind it.

Business explainer prepared from the Cognous Governance Evidence Pack whitepaper (de Lima, 2026).

1. Executive Summary

Cognous Governance Evidence Pack (formerly "Cognous Agent Governance Evidence Pack") is an open-source reference format, validator, traceable importer and Markdown renderer for business-facing review of AI-agent governance records. It helps teams summarize, in one structured artifact, what an agent is deployed to do, which systems it can access, which actions it can propose, which controls are declared, what was blocked, what sources the agent relied on, which reconstruction records exist, what validation found, what exports were redacted or signed, what risks remain open, and who reviewed the evidence.

Since the first edition, the component has gained something more important than new fields: a **traceable import path**. An evidence pack can now be generated from a declared Action Manifest and a retained Replay Bundle reconstruction, with every summary linked back to its source records, the exact producer revisions that created them and the version of the transformation that produced the pack. The importer also keeps different strengths of evidence apart: a declared control is not an implemented one, an implemented one is not a tested one, and none of these is an independently audited one.

It remains deliberately limited in what it claims to be. It does not run the agent. It does not enforce policy or grant authority. It does not certify compliance. It does not replace security, audit, legal or governance processes. What it does is create the structured artifact those processes can inspect: a single reviewable package where governance evidence is otherwise scattered across logs, tickets, spreadsheets and technical records.

The central message for enterprise leaders is this:

As AI agents move from demonstrations toward governed deployment, technical logs and runtime traces are not enough. Reviewers need an evidence package that explains what the agent is for, what it can do, what controls exist, what happened, what remains unresolved and who reviewed it, and that never claims more certainty than the underlying records support.

The distinction is simple. A Replay Bundle reconstructs what was recorded during a run; it answers the technical question. An evidence pack helps a business reviewer understand the deployment; it answers the governance question. Risk officers, auditors, security reviewers, legal teams and executive sponsors rarely read runtime traces. They read evidence. The Governance Evidence Pack gives them a structured, validated, renderable artifact built for that review, and it keeps the trail back to the records open for anyone who wants to check.

2. The Business Problem

AI agents are beginning to:

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

This creates a governance evidence gap. Runtime systems may produce detailed records: action proposals, authorization decisions, execution attempts, destination observations, reconstruction bundles, validation results, redaction metadata and signatures. Those records are necessary. But a deployment review board does not consume raw runtime records. Business reviewers need a coherent package, organized around deployment review rather than around technical events.

That gap shows up as seven recurring pain points.

1. Runtime records are too technical for business review. Traces, logs and reconstruction records are structured around events, identifiers and payloads. They serve developers and incident responders well, but executives, auditors, risk teams and business owners need to understand purpose, controls, risks and decisions, not event streams.

2. Evidence is scattered across systems. Relevant evidence may live in action manifests, control-plane records, executor records, reconstruction bundles, validation reports, redacted exports, risk registers, approval notes and manual documentation. Without a common package, every review begins with someone assembling the story by hand.

3. Governance reviews lack a stable artifact. Many reviews run on spreadsheets, slide decks, meetings, ticket comments or policy checklists. These are hard to validate, hard to compare across deployments, hard to render consistently and hard to keep current as the deployment changes.

4. Blocked and unresolved behavior is often invisible. A well-governed deployment is not only one where nothing bad happened. It is one where inappropriate actions were proposed and stopped, and where uncertain outcomes were recorded as uncertain. Reviewers need to see what was attempted, what was held or denied, and what is still unknown, not just what succeeded.

5. Source reliance is hard to summarize. Agents rely on documents, databases, APIs, tools, user input and model output. Technical reliance records may exist run by run, but reviewers need the aggregate: which sources, how often, for what scope, and where the supporting records live.

6. Risks and review decisions are detached from runtime evidence. Risk registers and approval decisions usually live in separate systems from the evidence that supports them. A reviewer approving a deployment should be able to see open risks and runtime evidence in the same artifact as the decision itself.

7. Summaries quietly inflate assurance. This is the pain point the first edition understated. When records are summarized, "declared" easily becomes "implemented", "the importer validated it" becomes "it was tested", "no acknowledgement" becomes "failed", and "an earlier approval exists" becomes "the action is permitted now". Each shortcut makes a report easier to read and less true. A business-facing package is only useful if it resists that drift.

These pain points are concrete. Consider four common deployments:

- A customer-service agent whose unauthorized outbound email attempts were blocked: evidence a reviewer should see, not lose.
- An internal research agent whose summaries depend on document sources and whose exports are controlled.
- A procurement agent whose purchase-order actions carry approval conditions.
- A refund workflow in which the caller timed out before the destination acknowledged, so nobody yet knows whether the refund was applied.

In each case, the technical evidence may exist. The business question is whether anyone can review it, as a package, in business language, with risks, decisions and uncertainty attached.

3. What the Governance Evidence Pack Is

The Governance Evidence Pack is a structured, business-facing evidence package for an AI-agent deployment. It summarizes deployment context, tool inventory, action inventory, authority model, policy controls, blocked actions, reliance sources, reconstruction bundles, validation results, redacted exports, risks and review decisions, with evidence references pointing from each summary back to the underlying records. When the pack is generated by the importer, it also carries a provenance section recording exactly which inputs, producer revisions and transformation produced it.

In one sentence:

It gives enterprises a structured, traceable answer to the question: "What evidence do we have about how this agent deployment is governed, and how strong is that evidence?"

The phrase "evidence pack" is worth translating into business terms. A governance evidence pack is:

- **a review artifact:** produced for deployment review, governance review, incident review or periodic audit;
- **a business-readable summary:** organized around the questions reviewers actually ask;
- **an index into technical evidence:** every summary can reference the records behind it;
- **a structured package for risk, security, audit and governance review:** validated, versioned and renderable;
- **derivative evidence:** a summary of other records, not an independent confirmation of what happened;
- **not a compliance certificate:** review status tracks review posture; it does not certify obligations;
- **not a permission:** a valid pack, a matching digest or a signature on its inputs never authorizes an agent to act;
- **not runtime enforcement:** it summarizes what upstream systems recorded; it does not gate anything.

Where it fits: the Cognous Open Control Stack

The first edition described a four-layer pattern ending in an "Evidence" layer. The stack has since grown. The Cognous Open Control Stack now comprises 13 components organized into six responsibility groups, plus a cross-cutting exchange integration:

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
Cross-cutting	Governed Exchange (GAX/IMX)

The groups are a responsibility map, not an execution sequence; nobody has to install all of them. The Governance Evidence Pack sits in group 05 alongside two close neighbours:

- **Replay Bundle** reconstructs retained Control Plane and executor records into a portable Reconstruction Bundle (format 0.2.0) without rerunning or re-authorizing anything.
- **Governance Evidence Pack** combines that reconstruction with the declared Action Manifest (version 1.1) and translates it into a business-facing package.

- **ODES** carries decision evidence to an outside recipient, who evaluates it under its own policy.

In the stack's one supported workflow, a synthetic refund, the reference run consumes the retained Replay, ODES and exchange artifacts for the transported operation and produces a Governance Evidence Pack, checking effect content and count, decision/effect/attempt continuity and original artifact commitments along the way. Hand-authored packs remain fully supported: a pack can still be assembled manually, or from other tooling, where no automated import exists.

4. What It Does

Summarizes the agent and its business purpose

The evidence pack identifies the agent, its business purpose, owner, business unit, user population and lifecycle stage.

Business value: reviewers know what is being evaluated and who is responsible for it before any technical detail appears.

Summarizes deployment context

The pack records the environment, deployment name, systems touched, data domains, geographic scope and notes. Production deployments carry stronger evidence expectations than sandboxes, and the validator issues extra warnings for production packs with weak review posture.

Business value: risk review is grounded in where the agent operates and what systems and data it touches.

Summarizes the tool and action inventories

The pack lists the tools available to the agent (with data classification, external-system linkage, access mode and control status) and the actions the agent can propose (with action type, tool, authority requirement, review requirement, reliance requirement, default posture and control status). When imported, these inventories are seeded from the declared Action Manifest.

Business value: security and governance teams can see the agent's system exposure and identify privileged or sensitive actions without reading code.

Summarizes the authority model and policy controls

The pack describes how privileged actions are governed (scopes, privileged action types, grant expiry, human approval) and records each policy control with an implementation status of implemented, partial, planned or not applicable, plus an evidence reference. An imported pack labels the Manifest's declarations as declarations: "planned", not "implemented", until other evidence says otherwise.

Business value: controls become visible and traceable, including the ones that are only declared.

Summarizes blocked actions and source reliance

The pack includes blocked-action summaries (action, type, tool, count, reason, evidence reference) and aggregates the sources the agent relied on (name, type, count, scope, evidence reference).

Business value: blocked behavior becomes evidence that controls operated, and reviewers can assess source grounding without reading run-level records.

Inventories reconstruction bundles and validation results

The pack lists available Replay Bundles or run records with status, generation date, signed and redacted flags, validation status and references, and records counts of valid and invalid bundles, warnings and errors.

Business value: incident review and audit can trace from the business summary down to technical reconstruction, and see whether the supporting records were internally consistent before relying on them.

Traces imported evidence to its sources (new)

The `agep import` command builds a pack from a Manifest v1.1 artifact and a Reconstruction Bundle 0.2.0 artifact. Every imported pack records the transformation version, the canonicalization profile, the identity, version and SHA-256 digest of each input, the pinned producer revisions, the source-record references and every import finding, including information lost in conversion. Source-supplied commitments and locally computed commitments are kept apart.

Business value: a skeptical reviewer can follow any summarized decision or outcome back to the exact record and producer revision behind it.

Keeps lifecycle facts separate (new)

The imported lifecycle summary distinguishes authorization granted or held, execution attempted, acknowledgement received or unknown, destination observed (absent, applied, partial, unknown or unavailable), retry permission and independent verification. A lost acknowledgement followed by an applied observation keeps both facts. Missing observations are recorded as unavailable, never as zero, failure or success. Historical authorization is labeled as historical: it does not establish current permission.

Business value: the report tells reviewers what is actually known, including what is not, instead of compressing a messy run into "success" or "failure".

Separates levels of assurance (new)

Imported packs grade control evidence on separate levels: declared, implemented, semantically validated during import, source-asserted runtime evidence, attributable test-run evidence, tested, operationally observed and independently audited. In the reference examples the later levels are "unavailable" by default, because no one supplied that evidence.

Business value: the pack makes the strength of the evidence as visible as its content, so a reader cannot mistake a synthetic import for an audit.

Maintains a risk register and records review decisions

The pack includes known risks with severity, status, mitigation, owner and evidence reference, and captures each review: reviewer, role, timestamp, decision and notes. The validator enforces a minimal

consistency rule: a pack cannot carry full approval while a critical risk remains open. An imported pack always starts in draft; software cannot create a human approval.

Business value: open risk and governance decisions sit alongside the evidence that informed them.

Renders a business-facing report

The renderer turns a pack into a Markdown report with sections for executive summary, deployment context, inventories, controls, risks, review records and known limitations. A traceable rendering adds a sixteenth section, "Traceability and Import Provenance", with input artifacts, derived counts, lifecycle status, evidence levels, findings and the complete escaped trace. The rendering is a deterministic presentation of the underlying data, not a separate source of truth.

Business value: the same structured data can be read by non-technical stakeholders without anyone rewriting it into slides, and without losing the material trace.

5. Why It Is Valuable

The benefits below are what the format is designed to support. They describe intended use, not measured outcomes: no enterprise deployment, time saving or reduction in review effort has been measured for this reference.

Better governance review. Reviewers receive one structured, validated artifact instead of scattered logs, spreadsheets, tickets and notes. Review becomes inspection of a package rather than assembly of a story.

Better executive readability. The pack translates technical runtime evidence into business language (purpose, exposure, controls, blocked behavior, risks and decisions) that a sponsor or board can read.

More honest assurance. Because declared, implemented, tested, observed and audited are recorded separately, a reader can see at a glance which claims rest on independent evidence and which rest on declarations or source assertions.

Better audit preparation. Audit teams can inspect a package that links deployment context, controls, reconstruction bundles, validation results and review decisions, with digests and source references pointing to the underlying records.

Better risk visibility. Open risks, mitigations, owners and review decisions appear alongside the runtime evidence, and the validator rejects full approval while a critical risk remains open.

Better incident reconstruction. When something goes wrong, the pack points from business summary to retained records, including unknown acknowledgements and unresolved outcomes, so investigators move from question to record without rebuilding the trail manually.

Better cross-functional alignment. Security, governance, engineering, product, legal, compliance and executives review the same artifact instead of each maintaining a partial picture.

Better deployment discipline. The existence of an evidence pack creates a structured checkpoint before or during deployment. If the pack cannot be filled in, that gap is itself a finding.

6. How It Is Different from Other Approaches

Enterprises already have several artifacts and systems that touch agent governance. The Governance Evidence Pack overlaps with none of them exactly.

Versus logs

Logs record events. They are indispensable for operations and forensics, but they do not summarize governance posture, controls, risks, review decisions or business context.

Difference: logs show technical events. The evidence pack organizes governance evidence for business review.

Versus Replay Bundles

Replay Bundles reconstruct retained technical records: what was proposed, decided, attempted and observed in a specific operation. They are the evidence pack's main runtime input; the pack inventories and references them, and its importer requires them to pass the accepted Replay semantic validator.

Difference: Replay Bundles answer "what was recorded in this run?" Evidence packs answer "what evidence supports this deployment review, and how strong is it?"

Versus ODES decision-evidence packages

ODES packages carry decision evidence across an organizational boundary so a recipient can evaluate it under its own policy.

Difference: ODES serves an external recipient deciding whether to rely on a decision. The evidence pack serves an internal review of a deployment.

Versus dashboards

Dashboards visualize data. They are views over records, and only as reviewable as the records beneath them.

Difference: a dashboard is a view. The evidence pack is a structured evidence artifact that a dashboard could display.

Versus GRC checklists

Governance, risk and compliance checklists ask whether controls exist. They capture assertions, often self-reported, without connecting them to runtime evidence.

Difference: checklists capture assertions. Evidence packs connect assertions to runtime records, reconstruction bundles, validation results, exports, risks and review records, and label assertions as assertions.

Versus policy documents

Policy documents define organizational intent. They say what should be true for any agent, not what evidence exists for a specific deployment.

Difference: policy documents say what should happen. Evidence packs summarize what evidence exists for one deployment.

Versus AI governance platforms

Governance platforms manage programs: inventories, approvals, risk assessments, reporting. That scope is complementary.

Difference: governance platforms manage the program. The evidence pack supplies a portable, per-deployment evidence artifact such platforms could consume.

Comparison at a glance

Category	What it does	What it misses	How the Evidence Pack differs
Logs	Record technical events	Business context, controls, risks, review decisions	Organizes governance evidence for review, with references to records
Replay Bundles	Reconstruct retained run records	Deployment purpose, risk posture, review decisions	Summarizes the deployment and traces back to the bundles
ODES packages	Carry decision evidence to a recipient	Internal deployment review context	Serves internal deployment review
Dashboards	Visualize data	A structured, portable, validatable artifact	Is the artifact a dashboard could display
GRC checklists	Capture control assertions	Linkage to runtime evidence	Connects and grades assertions against records
Policy documents	Define organizational intent	Deployment-specific evidence	Summarizes evidence for one deployment
AI governance platforms	Manage programs	A portable per-deployment evidence format	Supplies an artifact such platforms could consume

7. What It Is Not

Clear scope is part of the design. Cognous Governance Evidence Pack is not:

- an agent framework;
- a model runtime;
- a runtime control plane or enforcement layer;
- a source of authority or permission;
- a replay viewer;
- a hosted dashboard;
- a compliance certification product, legal advice or an independent audit;
- a complete enterprise governance platform;
- a security boundary by itself;
- a substitute for application authorization;
- a guarantee that model outputs are correct;
- proof that an external effect, such as a refund, was actually settled;
- a guarantee that regulatory obligations have been satisfied.

Its value is focused: it summarizes available evidence for review. The completeness and quality of that evidence depend on the upstream systems and governance processes that produce it. A blocked-action summary is only as good as the runtime records behind it; a risk register is only as complete as the process that maintains it; a review status is a package status, not a certification. A successful import does not show that a test ran, that an institution authorized current reliance or that a control is effective in production. The evidence pack makes gaps visible; it does not fill them.

A valid signature, chain inclusion, message receipt, reasoning instruction or evidence-package digest never authorizes execution. Authority must be supplied and evaluated through the appropriate trusted boundary.

8. Example Scenarios

A deployment review

Consider an enterprise preparing to review a customer-service agent before production. The agent reads CRM records, searches account notes, drafts customer emails and may attempt outbound sends only under approval. A hand-assembled evidence pack for that review would:

1. identify the agent: its name, business purpose, owner, business unit and target environment;
2. list the systems touched (the CRM, the notes index, the email gateway) and the customer-data domains involved;
3. list the tools and their data classifications, showing which link to external systems;
4. list the actions (CRM read, notes search, email draft, email send), each classified by action type, with email send flagged as requiring both authority and review;
5. describe the outbound-send authority scope, its approval expectation and its expiry posture;
6. record the review gates and blocked-send controls, each with implementation status and an evidence reference;
7. show that unauthorized email-send attempts were proposed and blocked, with counts, reasons and references;
8. summarize the sources relied on, the run records behind the summaries and whether those records validated;
9. show what was prepared for compliance review and which fields were redacted for which recipients;
10. list open and mitigated risks with severity, mitigation and owner;
11. record the reviewer's decision, for example approval with conditions, with the conditions stated.

The repository's customer-service example is a smaller, hand-authored version of this pack: two tools, two actions and three blocked email-send attempts, still `in_review`, which validates with a single warning that no review record exists yet. That warning is the format doing its job.

The synthetic refund with a lost acknowledgement

The stack's supported workflow shows the import path. A synthetic refund agent proposes a refund; the Control Plane authorizes it after checking current authority; the executor attempts the effect against a synthetic SQLite destination; the caller times out, so the acknowledgement is unknown. Later reconciliation, after an earlier observed-absent check, observes the effect as applied.

The repository keeps an imported pack for this "lost acknowledgement" case, generated from an earlier producer pair under transformation 0.3.0 (not the selected merged pair). It records:

1. one proposal, one decision, one distinct effect, and one Control Plane attempt and one executor attempt, kept in separate namespaces;
2. authorization: granted, labeled as historical evidence, with current permission "not evaluated from historical records";
3. acknowledgement: unknown; destination observed: applied; both facts retained;
4. reconstruction complete, which the pack states does not mean the effect completed;
5. retry permission: not established; independent verification: unavailable;
6. control evidence: declared (from the Manifest) and semantically validated during import, with tested, operationally observed and independently audited all unavailable;
7. review status draft, environment other, and unresolved issues stating that synthetic import success cannot support operational effectiveness or deployment approval.

The business outcome in both cases is the same: the review board receives one structured package. It can see what the agent is for, what it can do, what controls exist, what was blocked, what is still uncertain, what evidence supports the review and how strong that evidence is. Because every summary carries an evidence reference, a skeptical reviewer can drill from any claim down to the record behind it.

9. Why This Matters Now

Enterprises are exploring the move from AI experimentation to operational agent deployment, and the risk surface changes when agents can access tools, systems, data and workflows. What also changes, and is easier to miss, is the audience for evidence. In the demo phase the audience is the engineering team. In deployment it is risk committees, auditors, security reviewers, legal teams, customers and executives. Those audiences do not consume runtime traces.

A useful framing for leadership:

- **Chatbots require output review.** The text is the risk surface.
- **Agents require runtime control and replay.** Actual behavior must be governed, recorded and reconstructable.
- **Agent programs require evidence packages.** Deployment decisions need structured evidence, not tribal knowledge.
- **Evidence packages require honest grading.** A summary that turns declarations into facts is worse than no summary.

Technical records are necessary, but governance decisions require business-facing evidence that is traceable and does not overstate itself. The evidence pack is where retained records, risks and decisions meet in one artifact.

Teams that build evidence packaging into their review process before scaling agent deployment can answer audit, risk, legal, security and customer questions from an artifact that already exists. The format is designed so that evidence exists before reviewers ask for it.

10. Open-Source Reference Implementation

The Governance Evidence Pack is published as a public, open-source reference implementation at github.com/cogno-us/cognous-governance-evidence-pack (formerly `cognous-agent-governance-evidence-pack`). Teams can inspect the format, run the examples, validate packs, render business-facing reports and build compatible evidence workflows before making any platform commitments.

The repository includes:

- JSON Schema files defining the evidence-pack format, so other systems can produce and consume packs;
- a Python package (`agent-governance-evidence-pack`, Python 3.11 or later) with a bounded public object model built on Pydantic;
- a semantic validator that flags inconsistencies as errors or warnings, and that, for imported packs, re-runs the accepted Replay validator and fails closed when it is unavailable;
- the traceable importer for Manifest v1.1 and Reconstruction Bundle 0.2.0 inputs from supported producer revisions;
- a Markdown renderer, including the traceable rendering;
- the agep command-line interface with `validate`, `summarize`, `render`, `import` and `check-examples`;
- hand-authored example packs (customer service, internal research, procurement, finance) and imported synthetic-refund examples (lost acknowledgement, partial delivery, restart, prior absence and a rejected wrong-effect observation);
- test suites and dedicated qualification workflows run through GitHub Actions on Python 3.11 and 3.12;
- an Apache-2.0 license.

A first look takes four commands in a fresh virtual environment:

```
python -m pip install -e ".[dev]"
agep validate examples/customer_service_agent_evidence_pack.json
agep summarize examples/customer_service_agent_evidence_pack.json
agep check-examples
```

To see the component working with the rest of the stack, use the hub quickstart's clean-checkout verification for the current selected lock (PV-3) at github.com/cogno-us/cognous-open-control-stack, which checks out the locked revisions and runs two isolated repetitions.

Status and evidence

The stack as a whole is a merged, pinned **bounded synthetic reference**: one synthetic refund workflow, a synthetic SQLite destination, one host. The hub's component lock (profile merged-producers-v1) selects the Evidence Pack revision that adds the versioned agep-manifest-reconstruction-import/0.3.2 mapping for the merged Control Plane and executor pair; the imported pack schema remains 0.2.0, and earlier transformations (0.3.1, 0.3.0, 0.2.6) remain available for their original producers. Newer merged work, such as an optional read-only recovery-inspection transformation, exists in the repository but is not part of the selected release.

The current release evidence resolved all 35 acceptance scenarios, with required suites executed twice. A fresh reproduction on 13 October 2026 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-level totals, not a per-component count, and an older 915-test figure belongs to an earlier, historical evidence generation. Feature intake was frozen for release closeout on 13 October; public-preview approval is pending, and final publication and operational trust remain on hold.

11. Strategic Summary

The Governance Evidence Pack gives enterprises a practical way to convert AI-agent runtime governance records into business-facing evidence. It creates structure where evidence is otherwise scattered across logs, reconstruction bundles, validation reports, redacted exports, risk registers and review notes. It helps security, risk, compliance, audit, legal, business and executive stakeholders review agent deployments using a shared artifact, one that can be validated for consistency, rendered for readability, traced back to the exact records behind it and read with its strength of evidence in plain view.

It does not claim to certify compliance, grant authority, prove settlement or be a governance platform. It claims something narrower: that governance decisions deserve structured evidence, that the evidence should exist as a portable artifact rather than an assembly job repeated for every review, and that the artifact should never say more than its sources support.

Before enterprises can scale governed AI agents, they need more than runtime records. They need business-facing evidence packages that stay traceable and honest about what they do not know. That is the role of the Governance Evidence Pack.