PV-PP Agent Governance Demo Case

Wire Transfer Exception Review

Version 0.1 - Demonstration Case for the Agent Governance Runtime

Document Status

This document is a demonstration case, not a production specification, legal opinion, banking-control manual, or compliance certification.

The case is designed to show how the PV-PP Agent Governance Runtime can sit above an ordinary AI agent and tool layer to decide what actions are viable, not merely what actions are available.

Primary folder: 30 Applications and Extensions / Agent Governance / Demo Cases.

Table of Contents

1. Executive Summary

This demo case applies the PV-PP Agent Governance Runtime to an AI agent that reviews outgoing wire-transfer exceptions. The agent is assumed to classify failed auto-validation cases as routine, document-deficient, fraud-risk, compliance-risk, sanctions-risk, or senior-operations-risk. The core point is not that the agent releases funds. The core point is that an agent label can become de facto authority when human reviewers are overloaded, time pressure is high, and operational metrics reward fast clearance.

The ordinary orchestration question is: which tools can the agent call, and what output should it produce? The PV-PP question is stricter: under the current pressure, evidence quality, memory state, and governing-domain constraints, which actions remain viable without collapsing fraud control, sanctions integrity, auditability, client protection, or recovery capacity?

Demo Thesis

An agent can be technically non-authoritative while becoming operationally authoritative.

A governance layer must therefore evaluate the viability of the agent action, not merely the formal permission attached to the agent.

The Runtime converts tool availability into bounded action eligibility through corridor checks, memory/state checks, escalation requirements, and execution constraints.

2. Scenario Definition

A financial institution uses an AI agent to review outgoing wire transfers that fail automatic validation. The agent reads transaction data, client history, callback notes, approval records, invoice/supporting documentation, sanctions-screening results, fraud signals, and exception history. It then assigns a workflow outcome.

2.1 Baseline Agent Outputs

- Ready for release - routine exception cleared.
- Hold for documentation - missing invoice, callback, approval, deal support, or other required support.
- Escalate to fraud - possible business-email-compromise, account takeover, payment redirection, unusual beneficiary, suspicious timing, or inconsistent callback evidence.
- Escalate to compliance/sanctions - sanctions hit, incomplete screening, high-risk jurisdiction, AML concern, or unresolved ownership issue.
- Escalate to senior operations - operational edge case, cutoff conflict, client pressure, material amount, or unresolved cross-domain conflict.

2.2 Operating Pressure

Operations leadership tracks average exception-clearance time, wires released before cutoff, number of fraud escalations, manual review hours, and client complaints about delayed wires. These metrics create pressure for the agent to clear exceptions rapidly and reduce escalations. The agent may appear successful if it reduces delay and escalations while quietly increasing fraud exposure or weakening control evidence.

2.3 Workflow Reliance

Formally, every ready-labeled wire still receives human review. Practically, near cutoff, staff may rely on the agent label unless a case looks visibly abnormal. This creates human-in-loop laundering: a provisional machine output becomes operationally final through deadline pressure and reviewer fatigue.

3. Ordinary Orchestration Failure

A standard agent orchestration layer can register tools, check permissions, call retrieval systems, summarize evidence, and output a classification. That is insufficient because the highest-risk failure is not tool access failure. The highest-risk failure is viable-action failure: the agent has enough technical access to generate an answer, but the evidence state, control state, or recovery state does not support that answer.

Ordinary Layer	What It Can See	What It Misses
Tool registry	Which APIs and databases are available.	Whether using them is sufficient for a viable clearance decision.
Permission model	Whether the agent may access notes, case files, and screening data.	Whether the action should be blocked because evidence quality is too weak.
Prompt/rules layer	Static rules such as amount thresholds or required documents.	Dynamic pressure, inconsistent callback evidence, degraded staffing, and false-success incentives.
Scalar risk score	A single composite risk level.	Non-substitutable governing-domain failures such as sanctions uncertainty or missing callback confirmation.
Human review	A formal approval step.	Whether human reviewers are likely to rubber-stamp under cutoff pressure.

4. PV-PP Runtime Insertion Point

The Runtime sits between the agent planner/tool layer and operational execution. It does not replace the underlying agent. It governs the action proposed by the agent by asking whether the proposed output preserves required viability corridors.

Simplified flow:

- MCP/tool registry returns what is available.
- The agent proposes a classification or workflow action.
- The PV-PP Runtime evaluates whether that proposed action is viable under current governing-domain conditions.
- The Runtime permits, restricts, transforms, escalates, holds, or blocks the action.
- Execution proceeds only under the permitted authority boundary and logging requirements.

Available vs Viable

Available: the agent can call the callback-note system, document repository, client profile, sanctions tool, and fraud-history store.

Viable: the evidence returned by those tools supports the proposed action without violating governing-domain thresholds or disabling recovery.

5. Governing Domains

For this case, the Runtime should not use a single risk score as the governing primitive. It should preserve several non-substitutable domains. A gain in processing speed cannot compensate for a collapse in sanctions integrity or fraud-control evidence.

Domain	Preserved Corridor
Client-funds protection	Avoid misdirection, business-email-compromise loss, account-takeover transfer, or unauthorized beneficiary

Domain	Preserved Corridor
	change.
Fraud-control integrity	Ensure suspicious patterns, beneficiary changes, callback weaknesses, and historical anomalies trigger appropriate escalation.
Sanctions/compliance integrity	Prevent release when screening is unresolved, incomplete, inconsistent, or materially uncertain.
Evidence sufficiency	Distinguish strong evidence from weak notes such as left VM, usual number, or client has approved similar wires before.
Operational continuity	Clear routine exceptions without disabling cutoff performance or creating blanket escalation overload.
Auditability and recovery	Preserve enough logs, reasons, evidence state, and intervention points to reconstruct and remediate errors.
Authority-boundary integrity	Prevent ready labels from becoming hidden approval when the system is formally advisory.
Client-trust preservation	Balance delay costs against irreversible loss, regulatory exposure, and post-event credibility damage.

6. Runtime Decision Pattern

The Runtime evaluates an agent proposal through a staged governance pattern. This is deliberately not a single score. The pattern preserves hard domain thresholds, checks evidence quality, identifies metric-capture conditions, and applies restricted execution modes when full clearance is not viable.

1. **1. Proposed action intake:** Receive the agent proposal: ready, hold, fraud escalation, compliance escalation, sanctions escalation, or senior-ops escalation.
2. **2. State reconstruction:** Build the current perceived state from transaction data, notes, client history, tool outputs, document evidence, cutoff pressure, and staffing state.
3. **3. Governing-domain check:** Identify which domains are active and whether any are near threshold or already below threshold.
4. **4. Evidence-quality classification:** Classify callback evidence, approval evidence, document evidence, screening status, and anomaly evidence.
5. **5. Memory/state check:** Use prior incidents, client change history, beneficiary history, weak-note history, and reviewer reliance patterns as governance-relevant state.
6. **6. Policy eligibility:** Determine which workflow actions are admissible: full ready, guarded ready, hold, escalate, senior review, or block.
7. **7. Adequacy check:** Ask whether the proposed action preserves the corridor long enough to recover if the interpretation is wrong.
8. **8. Execution constraint:** If permitted, attach authority boundary, required evidence references, logging, reviewer warning, and rollback/escalation hooks.

7. Tool Eligibility Gates

A Runtime does not merely ask whether a tool can be called. It asks whether the result returned by the tool is sufficient to support the proposed authority level. The same tool output may be adequate for a hold decision but inadequate for a ready decision.

Tool/Data Source	Allowed Use	Viability Gate	Failure Response
Callback notes	Read and classify callback evidence.	Confirmed authorized signer is not equivalent to left VM or usual number.	Hold or escalate; do not clear as ready.

Tool/Data Source	Allowed Use	Viability Gate	Failure Response
Sanctions screening	Check current screening status and unresolved hits.	Unresolved or stale screening blocks release regardless of low fraud score.	Escalate to compliance/sanctions.
Client approval records	Verify authority, timing, and matching transaction terms.	Approval must match amount, beneficiary, account, timing, and signer authority.	Hold for documentation or senior ops.
Beneficiary history	Compare to known beneficiaries and recent changes.	New or changed beneficiary near cutoff activates fraud corridor.	Escalate or guarded hold.
Invoice/deal support	Read support documentation.	Support must correspond to transaction and not merely resemble routine support.	Hold for documentation.
Fraud-history store	Retrieve client/account/payment anomaly history.	Prior fraud, attempted redirection, or account-takeover indicators increase guard level.	Fraud escalation or senior review.
Case-management log	Record classification and reasons.	Log must be sufficient for audit and recovery; missing reason blocks silent clearance.	Require reasoned output before execution.

8. Memory and State Checks

The demo case depends heavily on typed memory. The Runtime should not treat each wire as an isolated classification event. Prior client behavior, prior fraud attempts, prior weak callback notes, prior staff reliance, and prior tool failures are part of the governance state.

- Client beneficiary-change history.
- Prior business-email-compromise attempts or suspicious payment-redirection episodes.
- Prior callbacks with weak or ambiguous notes.
- Known stale client-contact records or callback-team understaffing.
- Prior agent false-ready or near-miss events.
- Reviewer bulk-acceptance patterns near cutoff.
- Historical sanctions-screening latency or unresolved-hit patterns.
- Known document-ingestion weaknesses, including scanned PDFs, attachments, and incomplete deal files.

9. Available vs Viable Examples

Case	Available Action	Why It Is Not Necessarily Viable	Runtime Output
Callback note says left VM.	Agent can mark callback attempted.	Attempted contact is not confirmed authorization.	Hold for confirmed callback; no ready label.
Known client, same vendor category, new beneficiary account.	Agent can retrieve prior vendor history.	Vendor similarity does not clear account-change fraud risk.	Fraud escalation or guarded senior review.
Sanctions screening delayed.	Agent can say no hit currently returned.	No returned hit is not the same as completed screening.	Compliance/sanctions hold.
Cutoff in 12 minutes.	Agent can accelerate classification.	Time pressure increases hidden-authority risk and reviewer reliance.	Require explicit human attestation or senior ops route.
High-value wire with	Agent can infer routine	Inference cannot substitute	Hold for documentation.

Case	Available Action	Why It Is Not Necessarily Viable	Runtime Output
incomplete invoice.	commercial payment.	for required support at material amount.	
Low historical fraud rate.	Agent can lower risk score.	Rare severe loss dominates; base-rate comfort cannot override active anomalies.	Escalate if governing anomaly present.

10. Sample Runtime Output

Illustrative case input:

- Wire amount: $2.4 million.
- Beneficiary account changed within 48 hours.
- Callback note: called usual number, left VM, client has approved similar wires before.
- Sanctions screening: no current match, but updated beneficial ownership screen pending.
- Cutoff pressure: 9 minutes remaining; commercial client has complained about delays.
- Agent proposal: ready for release - routine exception cleared.

PV-PP Runtime Determination

Runtime result: BLOCK READY LABEL; ROUTE TO HOLD + FRAUD REVIEW + COMPLIANCE CHECK.

Reason: callback evidence is insufficient, beneficiary account changed recently, beneficial ownership screen is pending, cutoff pressure increases hidden-authority risk, and the proposed ready label would collapse client-funds protection and auditability corridors.

Permitted output: hold for documentation and confirmed callback; fraud review required before release; compliance must resolve pending ownership screen.

Execution constraints: log evidence basis; flag senior operations; disable bulk-accept path; require named human attestation before any override.

11. Corridor Analysis

The following table shows how a single ready label can produce corridor failure even when the agent is not formally authorized to release funds.

Domain	Condition	Ready Label Effect	Corridor Status	Required Response
Client-funds protection	New beneficiary + weak callback.	Creates high misdirection exposure.	Below corridor.	Block ready; fraud escalation.
Compliance integrity	Beneficial ownership screen pending.	Premature clearance before completion.	Below corridor.	Compliance hold.
Evidence sufficiency	Left VM / usual number note.	Treats weak evidence as confirmation.	Below corridor.	Require confirmed signer callback.
Operational continuity	Cutoff in 9 minutes.	Pressure favors shortcutting controls.	Stressed but not decisive.	Senior ops visibility.
Auditability	Agent says routine exception.	Reason does not preserve evidence logic.	Below corridor.	Require structured rationale/log.
Authority boundary	Human review formally required.	Bulk-accept likely near cutoff.	Below corridor.	Disable silent reliance path.

12. Sample Audit Output

A compact audit report for the scenario would read as follows:

PV-PP Agent Audit Report - Summary

Risk rating: High.

Primary failure mode: false-ready classification under cutoff pressure.

Most exposed domain: client-funds protection.

Secondary exposed domains: compliance/sanctions integrity, auditability, evidence sufficiency, authority-boundary integrity.

PP vs PPP gap: the agent perceives routine client history and apparent support, but actual productive power is constrained by weak callback evidence, pending ownership screening, recent beneficiary change, and reviewer-reliance pressure.

Metric-capture risk: the agent can appear successful by reducing delays and escalations while increasing rare severe-loss exposure.

Recommendation: block ready labels when callback evidence is weak, beneficiary details changed recently, screening is incomplete, or human review is likely to become rubber-stamp approval.

13. Adoption Path

13.1 Phase 1 - Audit-Only Mode

Run the Runtime in shadow mode. It reviews the agent output, produces a viability determination, logs conflicts, and identifies cases where the agent proposed an action that the Runtime would have blocked or constrained. No operational action is changed during this phase.

13.2 Phase 2 - Advisory Guardrail Mode

The Runtime adds warnings and required reason codes for high-risk cases. It can require stronger evidence before a ready label appears in the workflow, but final release remains governed by existing human controls.

13.3 Phase 3 - Binding Workflow Controls

The Runtime blocks or transforms certain agent outputs before they reach the release workflow. Examples include converting ready into hold, requiring fraud escalation, requiring compliance clearance, or disabling bulk acceptance for specific cases.

13.4 Phase 4 - Runtime Governance Integration

The Runtime becomes a persistent runtime layer above agent planning and tool access. It governs action eligibility, escalation pathways, memory use, evidence sufficiency, audit traces, and post-event recovery logic.

14. Product Implications

This demo case supports the product logic of the Agent Governance Runtime. The product is not merely an auditor, scorer, router, or dashboard. It is a runtime viability layer that can be deployed above agents to control whether proposed actions preserve governing-domain corridors.

- The Agent Auditor is the diagnostic front end: it identifies failure modes and weak corridors.
- The Runtime is the runtime control layer: it constrains proposed actions before they become operationally effective.

- Benchmark-derived implementation patterns supply reusable mechanisms: adequacy gates, typed memory checks, guarded re-entry, authority-boundary controls, and trace discipline.
- The enterprise value proposition is prevention of hidden-authority failure, false-success optimization, and irreversible high-stakes errors under pressure.

15. Limits of the Demo

This document is a structured demonstration. It does not validate a real banking system, quantify loss probability, replace banking controls, define regulatory compliance, or certify any production agent. A production implementation would require institution-specific control maps, legal and compliance review, technical integration, validation data, model-risk governance, and operational testing.

The demo also does not claim that scalar risk scoring is useless. Scalar scores can remain useful as internal signals. The PV-PP claim is narrower and stronger: scalar scores should not be the governing primitive where non-substitutable domains can collapse independently.

16. Repository Placement

Recommended primary location:

- 30 Applications and Extensions / Agent Governance / Demo Cases

Cross-reference from:

- PV-PP Agent Governance White Paper v0.1
- PV-PP Agent Governance Runtime Product and Technical Roadmap v0.1
- PV-PP Framework Runtime Implementation Pattern Catalog for Agent/Tool Governance v0.1
- PV-PP Framework Benchmark-Derived Architecture Lessons and Implementation Patterns v0.1