WHITE PAPER

From Shadow Mode to Straight-Through

Evidence-based autonomy for insurance claims and banking disputes on Pega



Varun Vemulapalli

November 20, 2026

Executive Summary

Claims teams should widen AI autonomy only when they can show, from records, that the AI is safe for a specific claim type, and prove later which rule allowed it. This paper sets out a five-part method to do that on Pega.

1. **Keep a side-by-side record on every case.** In shadow mode, store the AI's recommendation, the person's decision and the evidence each used in the Pega case history.
2. **Split disagreements by direction.** An AI that is looser than the adjuster creates risk; one that is stricter creates cost. Give them separate limits.
3. **Set thresholds per claim type.** A single program-wide rate lets easy, high-volume types hide weak performance on hard ones.
4. **Size the evidence window.** By the rule of three, zero looser disagreements in 600 claims supports a looser rate below 0.5% at about 95% confidence.
5. **Store thresholds as versioned Pega decision rules.** Rule history records who changed each limit, when and why, and the case records which version was live.

Together these produce an examiner-ready file aligned with SR 11-7, the NAIC model bulletin and the NIST AI RMF, and make rollback as traceable as rollout.

1. Introduction: The Question Every Automation Program Will Be Asked

Sooner or later, an examiner, auditor or board member will ask a claims leader one question: *why did you let the system decide this kind of claim on its own, and when?* Most automation programs cannot answer it from records. The decision to widen straight-through processing (STP) usually lives in a slide deck, a meeting and someone's memory.

The companion paper, *AI-Driven Financial Claims Automation with Pega*, recommended running AI in shadow mode before granting autonomy, then widening STP one claim type at a time. This paper turns that recommendation into a method. It answers four practical questions:

1. What should be recorded on every shadow-mode claim?
2. Which disagreements between the AI and the adjuster matter, and how much?
3. How much evidence is enough to widen STP for a claim type?
4. How do you prove later which rule was in force, who set it and why?

The method grew out of a public discussion on LinkedIn after the first paper was published. It applies to insurance claims and banking disputes alike, and uses Pega Platform features that most claims teams already license. The method and its thresholds are the author's proposal; they are a starting

point to adapt with an institution's risk, compliance and model-validation teams, not a regulatory standard.

2. The Side-by-Side Record

Shadow mode only produces evidence if every claim leaves a comparable record of what the AI would have decided, what the person decided, and why each decided it. Without that record, shadow mode is a demonstration, not a test.

The record belongs on the Pega case itself, not in a separate analytics store. Pega already keeps a case audit trail showing which operators worked on an item, when, and what automatic processes ran (Pega, n.d.-a). Teams can add their own entries to that history through standard utilities such as AddHistory or the History-Add method, and can enable field-level auditing on the properties that matter (MyKnowTech, n.d.). Keeping the comparison on the case means the reviewer, the auditor and the examiner all read the same file.

Field	What it holds	Why it matters
Claim type and segment	The category the threshold applies to, such as auto glass or card-not-present dispute	Thresholds are set per type (Section 4)
AI recommendation	Approve, deny, partial amount, representment or escalate	The "would have decided" half of the comparison
AI evidence and confidence	Documents and data the AI relied on, and its confidence score	Shows whether the AI was right for the right reasons
Human decision	The adjuster's or analyst's final outcome and amount	The reference the AI is measured against
Human evidence and notes	What the person relied on, especially when it differed	Explains disagreements
Disagreement class	Agree, looser or stricter (Section 3)	Feeds the threshold calculation
Rule and model versions	The decision rule version and model version in force	Answers "which version was live?" (Section 5)
Later outcome	Complaint, reopening, recovery, fraud finding or audit result	Catches cases where the person, not the AI, was wrong

The last row matters. A disagreement does not automatically mean the AI erred. Some adjuster decisions are later reversed on complaint or audit, and a fair comparison has to allow for that. Section 4 treats adjuster-correct and AI-correct disagreements differently.

3. Directional Disagreement: Not All Errors Cost the Same

A single agreement rate hides the most important fact about a disagreement: which way it went. When the AI is **stricter** than the adjuster, the cost is time and customer friction. When it is **looser**, the cost is money paid that should not have been, or a consumer protection breached. These two should never be added together.

Disagreement	Example (insurance)	Example (banking dispute)	Main cost	How to treat it
Agree	Both approve the same amount	Both grant provisional credit	None	Counts toward evidence
AI looser	AI would approve; adjuster denies or reduces	AI would accept a chargeback the analyst rejects	Leakage, fraud loss, regulatory exposure	Strict limit: must stay near zero
AI stricter	AI would deny or escalate; adjuster approves	AI would reject a valid dispute the analyst grants	Delay, extra handling, complaints	Wider tolerance; tune over time
Different but equivalent	Same outcome, minor amount difference within tolerance	Same outcome, different reason code	Minimal	Define the tolerance in advance

Two refinements make the classes reliable.

- **Define "looser" and "stricter" for each claim type before shadow mode starts.** For a denial-versus-approval decision it is obvious. For amounts, set a tolerance band, for example within 2% or within a fixed dollar figure, and treat anything outside it as directional.
- **Watch the stricter class for fairness, not just cost.** A model that is stricter mainly for one group of claimants is a fairness problem, even if each case only costs time. The NAIC model bulletin expects insurers to manage the risk of unfairly discriminatory outcomes from AI systems (National Association of Insurance Commissioners, 2023), so stricter disagreements should be broken down by relevant segments during review.

Splitting by direction also changes what the review queue is for. Looser disagreements are risk events and go to a senior reviewer. Stricter ones are tuning data and go to the team that owns the model and rules.

4. Per-Claim-Type Thresholds and the Evidence Window

Widen STP one claim type at a time, and only when that type's own looser-disagreement rate is demonstrably low. A single program-wide rate is misleading because an easy, high-volume claim type can pull the average up and carry a hard, low-volume type over the line with it.

4.1 What "near zero" means in numbers

"Near zero" needs a number, and zero observed errors is not the same as a zero error rate. Statistics offers a simple tool for this: the **rule of three**. If an event is not observed in *n* independent trials, the true rate is below 3/*n* with about 95% confidence (Wikipedia, n.d.; Eypasch et al., 1995). Rearranged, it tells a claims team how many shadow-mode claims of a type they need before they can make a claim about the looser rate:

$$n \approx 3 \div p_{max}$$

Here p_{max} is the highest looser-disagreement rate the institution will accept for that claim type, and *n* is the number of consecutive shadow-mode claims with no looser disagreements.

Highest acceptable looser rate	Claims needed with zero looser disagreements (95% confidence)	At 100 claims a week
1%	300	about 3 weeks
0.5%	600	about 6 weeks
0.2%	1,500	about 15 weeks
0.1%	3,000	about 30 weeks

The table shows why per-type thresholds matter in practice. A high-volume type such as low-value card disputes can reach a strict bound in weeks. A low-volume, high-severity type such as life claims may never gather enough evidence for full STP, and that is a legitimate finding in its own right: it tells the institution to keep people in the decision.

4.2 Rules for the evidence window

- **Fix the window before you start.** Decide the claim count, minimum calendar time and acceptable rate in advance, and record them in the versioned rule described in Section 5. Moving the goalposts after seeing results undermines the evidence.
- **Reset the window after a material change.** A new model version, a decision rule change, a policy wording change or a card network rule release starts a new window for the affected claim types.
- **Use exact methods for small samples or nonzero counts.** The rule of three is a quick approximation for zero events. When looser disagreements are observed, or samples are small, a statistician should compute an exact binomial bound.
- **Count only reviewed disagreements.** If later evidence shows the adjuster, not the AI, was wrong, reclassify the case. Keep both the original and the reclassified label in the record.
- **Give the stricter class a separate, wider limit.** Stricter disagreements are tracked against cost and fairness targets, not the risk limit.
- **Choose a calendar window as well as a count.** A minimum period, for example one full month, captures seasonality and month-end patterns that a quick burst of claims would miss.

5. Thresholds as Versioned Pega Decision Rules

Every threshold that controls STP should live in a versioned Pega decision rule, not in a configuration value, a spreadsheet or code. That single choice turns the examiner's question, "who changed this, when and why?", from a search into a lookup.

5.1 Why a rule, not a setting

Pega records a history for each rule. From a rule's History tab, a user can see when it was changed and by whom, add a memo, compare historical versions and restore an earlier one (Pega, n.d.-b). When a ruleset is secured, a developer must document each change when editing a rule, which creates an auditable history of the ruleset (Pega Academy, n.d.-a). Rules are also grouped into numbered ruleset versions in a major-minor-patch format, which makes each release identifiable (Pega Academy, n.d.-b).

A plain configuration value has none of these protections by default. It can be changed quietly, and the reason is rarely recorded.

5.2 What the threshold rule should contain

The author recommends one decision table per claim family, with a row per claim type. Each row holds:

Column	Example value
Claim type	Card-not-present dispute under $100
Autonomy level	Shadow, supervised or straight-through
Maximum looser rate	0.5%
Maximum stricter rate	5%
Evidence window	600 claims and at least 30 days
Approved by	Named claims and model risk owners
Rationale and evidence reference	Link to the shadow-mode report for the window
Review date	Next scheduled review

The rule's history memo should repeat the approver and the evidence reference for each change, so the trail exists even if a linked document later moves.

5.3 Proving which version was live

The final link is on the case. When a claim is routed straight through, the case should record the version of the threshold rule and of the AI model that allowed it, as the side-by-side record in Section

2 already requires. An examiner reviewing any claim can then trace it to the exact rule version, the evidence window that justified it and the people who approved it.

The same design makes rollback simple. Restoring an earlier rule version narrows autonomy for a claim type immediately, and that change is recorded like any other.

6. The Examiner-Ready File

The method produces, as a by-product, the evidence that existing supervisory frameworks already ask for. No new paperwork is needed, only discipline about where the records live.

For banks, the reference point is the Federal Reserve's SR 11-7 guidance on model risk management. Its guiding principle is "effective challenge" of models by objective, informed parties. It expects validation to continue after a model goes into use, and describes back-testing as comparing actual outcomes with model outputs over a period not used to build the model (Board of Governors of the Federal Reserve System, 2011). Shadow mode is a form of exactly this outcomes analysis, with the adjuster's decision and later outcomes as the benchmark. For insurers, the NAIC model bulletin expects a written AI program that manages the risk of inaccurate or unfairly discriminatory decisions (National Association of Insurance Commissioners, 2023). The NIST AI Risk Management Framework supplies a common structure for both (National Institute of Standards and Technology, 2023).

What a reviewer asks	Where the method answers it
How was the AI tested before it acted alone?	Shadow-mode side-by-side records on each case (Section 2)
How do you know it performs well enough?	Looser-disagreement rate per claim type, with the evidence window and confidence bound (Section 4)
Are some errors treated as more serious?	Directional classes, with separate limits for looser and stricter (Section 3)
Who decided to widen automation, and why?	Versioned threshold rule with approver and evidence reference in its history (Section 5)
Which rules applied to this specific claim?	Rule and model versions stamped on the case
How do you detect deterioration?	Ongoing monitoring of disagreements on sampled STP claims, with rollback triggers (Section 7)
Is the outcome fair across customers?	Stricter-disagreement breakdown by relevant segments (Section 3)

One caution: vendor models are not exempt. SR 11-7 coverage includes models bought from third parties, so an AI agent supplied by Pega or another vendor still needs this evidence from the institution's own claims.

7. Operating the Method: Widen, Monitor, Roll Back

Each claim type moves through three autonomy levels, and can move back as easily as forward. The threshold rule from Section 5 records which level each type is at.

1. **Shadow.** The AI recommends on every claim; people decide everything. Side-by-side records accumulate until the evidence window is met. *Move forward when* the looser rate is within its limit over the full window, and the approvers named in the rule sign off.
2. **Supervised.** The AI acts, but a person reviews each decision, or a large sample, before it takes effect. This stage tests the full workflow, including payments, letters and integrations, with a safety net. *Move forward when* a second window confirms the shadow-mode result under live conditions.
3. **Straight-through.** The AI decides within the rule's limits with no review before the decision. A random sample of completed claims is still reviewed by people each week, so the side-by-side record continues. *Move back immediately when* any rollback trigger fires.

7.1 Rollback triggers

The author suggests setting these in the threshold rule for each claim type:

- A looser disagreement found in the post-STP sample that exceeds the limit set for that type.
- A complaint, regulatory finding or fraud case traced to an STP decision.
- A change to the AI model, decision rules, policy wording or card network rules affecting the type.
- Drift in the inputs, such as a new channel, product or customer segment appearing in volume.

Rolling back means restoring the earlier rule version so the claim type returns to supervised mode. Because this is itself a recorded rule change, the reason for every narrowing of autonomy is preserved as carefully as the reason for every widening.

7.2 Testing changes before they reach claims

Pega's decisioning tools already support this pattern elsewhere. In Customer Decision Hub, teams can run a challenger strategy against a champion in simulation and compare results without putting the challenger into production (Pega Academy, n.d.-c). They can also run simulations on a sample of production data in a separate environment before deploying a change (Pega, n.d.-c). Claims teams can borrow the same discipline: replay a recent window of claims through a proposed rule or model change, and compare its decisions with the current version before any shadow window restarts.

8. Conclusion

Shadow mode is widely recommended and rarely made rigorous. The method in this paper makes it rigorous with four habits: record the AI's call and the person's call side by side on every case; split

disagreements by direction; set per-claim-type limits backed by a stated evidence window; and keep those limits in versioned Pega decision rules whose history names who approved each change and why.

None of this requires new technology. It uses the case audit trail, rule history and ruleset versioning that Pega already provides, and it produces the outcomes analysis that supervisors already expect. The payoff is that widening automation stops being a judgment call made in a meeting and becomes a decision anyone can trace, defend and, when needed, reverse.

References

Board of Governors of the Federal Reserve System. (2011, April 4). *SR 11-7: Guidance on model risk management.* https://www.federalreserve.gov/supervisionreg/srletters/sr1107.htm

Eypasch, E., Lefering, R., Kum, C. K., & Troidl, H. (1995). Probability of adverse events that have not yet occurred: A statistical reminder. *BMJ, 311*(7005), 619–620. https://doi.org/10.1136/bmj.311.7005.619

MyKnowTech. (n.d.). *History writes and case auditing in Pega.* https://myknowtech.com/blog/code-vault/history-writes-and-case-auditing-in-pega

National Association of Insurance Commissioners. (2023). *NAIC model bulletin: Use of artificial intelligence systems by insurers* [Adoption map]. https://content.naic.org/sites/default/files/legal-adoption-map-ai-model-bulletin.pdf

National Institute of Standards and Technology. (2023). *Artificial intelligence risk management framework (AI RMF 1.0)* (NIST AI 100-1). https://doi.org/10.6028/NIST.AI.100-1

Pega. (n.d.-a). *Audit trail* [Pega Platform help]. https://docs-previous.pega.com/sites/default/files/help_v718/definitions/a/audittrail.htm

Pega. (n.d.-b). *Rule history* [Pega Platform help]. https://community.pega.com/sites/pdn.pega.com/files/help_v731/designer-studio/history/v6completingrulehistory.htm

Pega. (n.d.-c). *Sample, migrate, and test production data in a simulation environment (8.3).* https://docs.pega.com/whats-new-pega-platform/sample-migrate-and-test-production-data-simulation-environment-83

Pega Academy. (n.d.-a). *Rule updates in a secured ruleset.* https://academy.pega.com/node/22116

Pega Academy. (n.d.-b). *Rulesets in an application: Ruleset versioning.* https://academy.pega.com/node/72711

Pega Academy. (n.d.-c). *Introducing decisioning simulations.* https://academy.pega.com/pt-br/topic/introducing-decisioning-simulations/v1/in/9826

Vemulapalli, V. (2026). *AI-driven financial claims automation with Pega* [White paper].

Wikipedia. (n.d.). *Rule of three (statistics).* https://en.wikipedia.org/wiki/Rule_of_three_(statistics)

About this paper

Author: Varun Vemulapalli. **Published:** November 20, 2026.