

Operationalising CAPA for AI: Evidence Requirements and Limits from a Production Case Study

Musarat Kabir-Chisty

md@cloutechinnovations.com

Cloutech Innovations

Research Article

Keywords: CAPA for AI, AI governance, Root-cause analysis, Deviation management, Meaningful human oversight, Operational machine learning

Posted Date: October 14th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10893823/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Abstract

This paper argues that the mature quality-engineering disciplines developed for regulated manufacturing, namely deviation management, root-cause analysis (RCA), corrective and preventive action (CAPA), and change control, provide a rigorous, ethically motivated method for governing AI automation, and that an AI system's own operational telemetry can serve as the deviation record those disciplines require. We demonstrate the approach through a single in-depth case study of a live document-automation system in a regulated UK immigration-law practice, drawing on an operational log of 42,866 automated actions across 597 cases and 73,846 documents from a relaunch and migration period. Treating the log as a quality record, we construct a seven-class deviation taxonomy and rank it by Pareto analysis, finding that three classes (duplication, verification failure, and misclassification) account for 71.8% of 17,080 remediation events, while showing the ranking should be read at the incident level (2,717 incidents), where it re-orders. An Ishikawa analysis and three worked five-whys chains trace surface deviations to systemic factors that implicate knowledge transfer more than model capability. We reconstruct the corrective actions, preventive changes, and versioned deployments taken, showing that the deployment ran the CAPA cycle up to, but not including, formal effectiveness verification, which we identify as the key requirement for closing the loop. We discuss the ethics of treating AI failures as quality deviations, human deferral as designed oversight, and deploy-then-remediate practice. The contribution is a transferable governance method that operationalises accountability and meaningful human oversight for organisational AI, together with the evidence it needs to become fully closed-loop.

1. Introduction

Organisations increasingly delegate consequential clerical and decision-support work to AI systems, and the discourse on when these systems succeed or fail is dominated by two framings: a technical one (accuracy, calibration, benchmark performance) and a principled one (fairness, transparency, accountability), the latter articulated in trustworthy-AI frameworks and management standards [1–3]. Both are necessary, but neither tells an operational manager how to diagnose and remediate a system that is quietly producing the wrong outcome in production: how to distinguish a symptom from its cause, or how to know whether a fix worked.

A century of quality engineering exists to answer exactly these questions. In regulated industries, departures from expected behaviour are logged as *deviations*, investigated through *root-cause analysis* (RCA), remediated through *corrective and preventive action* (CAPA), and any resulting modification is governed through *change control*, with effectiveness verified before closure [4, 5, 6]. There is an active and useful body of practice on using AI *within* quality management. This paper concerns the less-developed inverse, which we call **CAPA for AI**, in which the deployed automation is itself the object under quality management, and its failures are the deviations under investigation.

We argue that this transfer is both possible and valuable, and we demonstrate it on a live deployment while being precise about the evidence it rests on. Modern AI deployments can be instrumented at fine granularity: each automated action, its confidence, its operator, and its outcome can be logged, and that log is a ready-made deviation record. The question we pursue is how well these disciplines transfer in practice: how far an AI system's telemetry can carry the cycle of deviation, prioritisation, RCA, CAPA, change control, and effectiveness verification, and what additional evidence is needed to complete it.

We answer through a single, in-depth case study of a live document-automation system in a regulated UK immigration-law practice, using its operational database as the quality record. We are explicit about the data's character, because it bounds the claims: it captures a relaunch and migration period rather than a long steady state, a large share of it is a single remediation campaign, and it is one organisation in one domain. The contribution is a transferable, cross-disciplinary governance method that treats AI-automation failures as quality deviations, demonstrated end to end on real operational data, together with a specification of the additional evidence, beyond telemetry alone, that a fully closed-loop quality system for AI requires. This stance is consistent with critiques of RCA and five-whys methods that caution against overstating causal certainty in complex systems [16, 17].

Section 2 positions the work against AI-governance, operational-ML, human-deferral, and systems-safety literatures, and motivates the choice of CAPA over alternatives. Section 3 describes the case system, the unit-of-analysis hierarchy, the deviation taxonomy, and the ethical guardrails on the data. Section 4 presents the findings, organised as an evidence hierarchy from what telemetry reliably observes to what it cannot establish. Section 5 discusses implications, including deferral as an active-learning signal and risk-weighted rather than count-weighted prioritisation. Sections 6 and 7 give limitations and conclusions. An appendix documents the deviation-class definitions for reproducibility.

2. Background and related work

2.1 Failure in operational AI is socio-technical

Production AI systems fail for reasons that lie largely outside the model. Sculley et al.'s account of hidden technical debt in machine-learning systems shows that the model is a small part of a large, fragile socio-technical apparatus of data pipelines, configuration, and glue code [9]; work on AI safety catalogues operational hazards such as distribution shift and unsafe exploration [29, 30]; and empirical studies of dataset shift show how silently such conditions degrade deployed models [28]. Sociotechnical critiques of purely technical framings reinforce the point that governance must attend to the system around the model [8]. Accountability instruments such as internal algorithmic auditing [11] and documentation practices such as model cards [12] provide important lifecycle accountability mechanisms, but do not by themselves supply the incident-to-investigation-to-corrective-action-to-effectiveness-verification chain examined here. Production machine-learning practice has meanwhile matured its deployment and monitoring methods [36] and begun cataloguing real-world failures in public incident databases modelled on aviation safety reporting [35]; what remains comparatively under-

developed is a disciplined, repeatable *operational* method that links deviation detection through investigation and corrective action to verified effectiveness once a system is live: the accountability gap this paper probes.

2.2 Deviation management, RCA, CAPA, and change control

In regulated quality systems a *deviation* is any departure from an approved procedure, specification, or expected outcome, classified by risk (commonly critical, major, minor) and investigated to distinguish *symptom*, *proximate cause*, and *root cause* [13, 14]; the human-error tradition further warns that systemic failures usually reflect latent organisational conditions rather than individual mistakes [20]. RCA tools such as the Ishikawa (fishbone) diagram, five-whys, Pareto analysis, and failure-mode-and-effects analysis structure the search [14, 19]. The confirmed root cause drives a *corrective action* (fixing the instance) and a *preventive action* (eliminating the systemic cause), whose effectiveness is verified before closure [5, 6]; where remediation alters the system, *change control* governs the modification. These practices are anchored in ISO 13485 [13], the FDA's device-quality regulation (historically the Quality System Regulation, now harmonised with ISO 13485 under the Quality Management System Regulation) [6], and ICH pharmaceutical-quality frameworks [4, 5], with a generic form in ISO 9001 [15]. The literature is candid about RCA's limits, which matters here: "root cause" language can imply a single linear cause where complex systems have interacting ones; investigations are often truncated; and five-whys can terminate prematurely [16, 17]. We treat our causal statements accordingly.

2.3 Why CAPA, and not only FMEA, STAMP/STPA, or Safety-II

CAPA is one of several operational safety families, and reviewers rightly ask why it in particular. Failure-mode-and-effects analysis is prospective and risk-ranking rather than incident-driven; systems-theoretic accident models such as STAMP/STPA reject linear root-cause reasoning in favour of control-structure analysis [31]; and resilience engineering (Safety-II) argues that safety cannot be understood through failure analysis alone but must study how work usually succeeds [32]. We do not claim CAPA is superior. We select it because it offers an unusually tight, auditable linkage from an observed deviation through investigation and action to a *controlled change* and a *verified effect*: the very linkage that AI-governance frameworks demand but leave abstract. We also adopt the systems-safety critique directly: our analysis treats causes as contributing factors within a control structure rather than as single roots, and Section 5 returns to what STAMP/STPA and Safety-II would add.

2.4 Deferral, selective prediction, and human oversight

A central design feature of the case system, automating when confident and deferring to a human otherwise, is the applied face of a substantial technical literature on selective prediction and learning to defer [25, 26, 27]. This work formalises the accept-or-defer decision and its cost trade-offs, and is directly relevant to how deferral telemetry should be interpreted. It connects, in turn, to accounts of

appropriate reliance and automation bias [21, 22], to guidelines for human-AI interaction that treat calibrated hand-offs as a design concern [10], and to accounts of meaningful human control [18]. We engage this literature both to interpret our deferral findings correctly and to identify a technical, not merely governance-level, contribution the data suggests (Section 5.2).

2.5 Positioning relative to AI-governance frameworks

The method proposed here is best read as an *operational implementation pattern* beneath existing frameworks rather than a replacement for them. The NIST AI Risk Management Framework 1.0's *Measure* and *Manage* functions call for ongoing measurement of and action on AI risk [2], with its subsequent Generative AI Profile giving more specific lifecycle risk-management guidance for generative systems [34]; ISO/IEC 42001 requires an auditable AI management system [3]; and ISO/IEC 23894 gives AI risk-management guidance [23]. Comparative analyses situate these instruments within a wider landscape of international AI-governance policy frames [24]. Most directly, for high-risk systems within its scope, Article 72 of the EU AI Act requires a documented post-market monitoring system that actively and systematically collects and analyses performance data throughout the system lifecycle [33]; the telemetry-to-CAPA pattern examined here can be read as one possible operational mechanism for generating part of that monitoring evidence, without implying that telemetry alone establishes compliance or that this particular system falls into any specific statutory risk category. Internal algorithmic auditing [11] provides an accountability framing; our emphasis differs in being incident-driven and recurrence-oriented. Model cards [12] are complementary, documenting the model, whereas most failures here arose outside it. The approach also instantiates the later stages of the AI Technology Transfer Framework (AITTF), namely validation, reproducibility, oversight, and continuous verification [7], while, as Section 4 shows, exposing the evidence those stages presuppose but that the deployment lacked.

3. The case system and method

3.1 The deployment

The case system is a live, locally hosted document-automation platform used by a regulated UK immigration-law practice to manage casework end to end. Documents enter through several channels and pass through a single pipeline: a security scan; case identification; optical character recognition; classification of each document by a large-language-model classifier; *slotting*, in which a document is auto-filed into a case's document slot if the classifier's confidence clears a threshold and is otherwise routed to a tiered human-review queue; an independent verification step assigning a red/amber/green verdict (green ≥ 0.85 , amber ≥ 0.55 , else red); human review; and finally filing, drafting, and submission. Of 42,866 logged actions, 40,700 (95%) were performed by automated pipelines and scripts. To convey scope without disclosing anything client-identifying, the deployment covered 597 cases and 596 clients across 12 destination countries and 18 visa categories, encompassing 73,846 catalogued documents spanning 191 document types in 11 categories, across 50,172 slots.

3.2 Data character: a compressed migration snapshot

Honesty about the timeline is essential because it bounds every downstream claim. Although the log's timestamps span 10 June to 18 August 2026, 41,929 of 42,866 actions (97.8%) fall in June, and the corrective sweeps each executed on a *single day* (coverage-gap deferrals on 14 June, re-slotting on 16 June, de-duplication on 17 June), with negligible activity thereafter. This is therefore a relaunch-and-migration snapshot, not a longitudinal record of steady-state operation. It is well suited to a retrospective deviation analysis and entirely unsuited to a before/after effectiveness study, a constraint we make analytic use of rather than paper over.

3.3 A unit-of-analysis hierarchy

A logged event is not a failure. One defect can generate many events; one event can affect many objects; batch jobs and retries multiply records; and some actions (e.g., a deferral) may be intended behaviour. We therefore define four levels explicitly and report counts at each where the data permits: **log event** (e.g., a re-slot operation), **affected item** (the document or slot moved), **incident** (a distinct case-level occurrence, operationalised as a unique case × deviation-type × day), and **root-cause episode** (a single underlying mechanism, e.g., a migration that omitted canonical slot codes). We do not equate events with deviations; Section 4.2 reports the sensitivity of prioritisation to this choice.

3.4 The deviation taxonomy

We mapped the log's action vocabulary onto seven deviation classes by inspecting the system's documentation and code to establish each action's operational meaning, grouping actions that represent *detection and remediation of a defect*: duplication, verification failure, misclassification/re-slot, coverage-gap deferral, data-integrity fix, mis-routing, and case-identity error. Normal automated placement, routine logins, and one-off historical backfill (approximately 27% of the log, dominated by a single large-client migration) were excluded, as were by-design behaviours (Section 3.5). The class-to-action mapping and the incident de-duplication rule are given in the Appendix so the counts are reproducible.

One class requires careful definition. A *deferral* (the filing agent declining to auto-place and escalating to a human) is, per the system specification, correct behaviour, not a departure from it. It is therefore not itself a deviation. What the deferral telemetry *surfaces* is a coverage deficiency: a document type the automation cannot place. We accordingly define the class as "coverage deficiency surfaced via deferral telemetry", and treat the deferral count as a signal of that deficiency, not as a count of erroneous system behaviour. This resolves an inconsistency that a naive "deferral = deviation" definition would introduce.

3.5 Analytical methods and ethical guardrails

We applied Pareto analysis for prioritisation, an Ishikawa diagram to organise candidate causes, and five-whys to trace representative deviations to candidate causal mechanisms, and we reconstructed the change history from the engineering record. Four honesty constraints shape the analysis. First, *by-design behaviour is excluded*: outbound email carried a "fallback" status in 613 of 619 messages, but the

documentation confirms this is the intended behaviour when the mail transport is unconfigured, so it is not a deviation. Second, *we report no failure rate without a denominator*: only failing verifier verdicts are logged, not the total verified, so we report red-verdict counts, not a “verification failure rate”. Third, *the data is a migration snapshot* (Section 3.2). Fourth, *causes are interpretive*: consistent with the RCA critiques [16, 17] and systems-safety view [31], our root-cause statements are evidence-grounded hypotheses with stated alternatives and confidence, not proven causes.

The data-handling ethics are strict. The underlying records contain confidential personal data; nothing at the level of an individual, case, or document is reported. All figures are aggregate counts and distributions; the organisation is described only generically; and the single large remediation campaign is referred to without any identifier. The analysis is an operational audit of a live production system by investigators with operator access; no third-party human-subjects research was involved. This access affords depth but is a source of potential hindsight bias, which we flag where it bears on interpretation.

4. Findings

We present the findings as an evidence hierarchy: what the telemetry reliably observes (4.1), how events become prioritised incidents and how fragile that prioritisation is (4.2–4.3), what causal weight the RCA can bear (4.4–4.5), and what the data cannot establish about thresholds and effectiveness (4.6–4.7).

4.1 What telemetry reliably observes

The log records 42,866 actions over the snapshot, 95% by automated components, confirming that the system is a genuine automation making and executing large numbers of filing and classification decisions with limited per-item human involvement, and that it is instrumented finely enough to observe those decisions and their corrections. Detection, the first CAPA stage, is therefore well supported: recurring, machine-readable failure patterns are present and countable.

4.2 Prioritisation is not robust to the unit of analysis

Consolidating the log into the seven classes yields 17,080 remediation events. Figure 1 presents these as a Pareto chart; by events, the three largest classes account for 71.8%: duplication (4,692; 27.5%), verification failure (4,206; 24.6%), and misclassification/re-slot (3,364; 19.7%). But this ranking is an artefact of counting events. Table 1 recomputes counts at the incident and case levels. The 17,080 events collapse to 2,717 case-level incidents (4.6 per case; 36.8 per 1,000 documents), and the leading classes re-order: by incidents the top three become misclassification (661), data-integrity (654), and duplication (611), while verification failure falls from second by events to fifth by incidents: its 4,206 events reduce to 169 incidents across 168 cases, i.e., many repeated flags on few cases. The prioritisation a manager would draw depends heavily on the unit chosen. This instability, rather than any single ranking, is the finding: frequency of logged events is a poor and manipulable basis for prioritising high-stakes AI deviations, and incident-level denominators are a prerequisite for defensible prioritisation.

Table 1

Deviation taxonomy with definitions and unit-of-analysis sensitivity. Incidents are unique case × deviation-type × day; rates use 73,846 documents.

Deviation class	Definition (remediation logged)	Events	Incidents	Cases	Events/1k docs
Duplication	Duplicate document/slot collapsed	4,692	611	401	63.5
Verification failure	Failing (red) verifier verdict	4,206	169	168	57.0
Misclassification / re-slot	Document moved to corrected slot	3,364	661	262	45.6
Coverage deficiency (via deferral)	Auto-placement declined; escalated	2,349	169	169	31.8
Data-integrity fix	Stale pointer / orphan / placeholder	1,532	654	304	20.7
Mis-routing	Document routed to wrong annex	846	399	199	11.5
Case-identity error	Duplicate / mis-keyed case merged	91	54	54	1.2
Total		17,080	2,717		

4.3 Frequency prioritisation diverges from risk prioritisation

Frequency is in any case the wrong sole criterion for a high-stakes setting that invokes quality-risk thinking [4]. Table 2 sets the classes in an ordinal risk view combining an investigator-assigned severity, detectability, and reversibility (these ratings are judgments, not measurements, and are offered to illustrate the divergence, not to fix priorities). The 91 case-identity errors, the rarest class by every count, plausibly carry the highest severity and lowest reversibility, since a merged or mis-keyed case can attach one person's documents to another's matter in an immigration context. A count-weighted Pareto would rank this class last; a risk-weighted view would elevate it. Any serious "CAPA for AI" must prioritise on risk, not frequency, and must therefore acquire severity information the raw telemetry does not carry.

Table 2

Investigator-assigned ordinal risk view (illustrative; severity/detectability/reversibility are judgments, not measured).

Deviation class	Frequency	Severity	Detectability	Reversibility	Indicative priority
Case-identity error	Low	High	Low	Low	High
Verification failure	High	Med–High	High	Medium	High
Misclassification / re-slot	High	Medium	Medium	High	Med–High
Duplication	High	Low–Med	High	High	Medium
Data-integrity fix	Medium	Low–Med	Medium	High	Medium
Mis-routing	Low–Med	Medium	Medium	Medium	Medium
Coverage deficiency (deferral)	Medium	Low (by design)	High	High	Low–Med (monitor)

4.4 Candidate contributing factors (Ishikawa)

Figure 2 organises candidate causes into six branches adapted from manufacturing to an AI-automation context. We label it, deliberately, as a *retrospective, investigator-generated hypothesis map*, not a diagram of established causes: it was produced by a single investigator with operational knowledge of the system, from the engineering record, after the fact, and is therefore exposed to hindsight bias. Its value is not proof but structure: specifically, that candidate causes distribute across all six branches rather than concentrating in Model and Method, and that several of the most consequential sit in Data and Inputs and Process and Workflow (image-only scans yielding blank text; imported slots lacking classification codes; template placeholders masking real files; inheritance gaps failing to map documents to the correct route). In AITTF terms these are failures to capture and transfer the workflow and its exceptions [7]; in STAMP terms they are inadequate constraints in the control structure [31]. Both readings are compatible; neither is established by the diagram alone.

4.5 Causal hypotheses, with alternatives and confidence

Table 3 replaces the language of “confirmed root cause” with explicit hypotheses, the evidence for each, competing explanations, and a confidence rating. The five-whys chains behind these entries are plausible narratives grounded in code inspection, database state, and the named-backup record, but they are under-determined: the same duplication mechanism that we read as a knowledge-transfer failure could as reasonably be labelled migration-schema incompatibility, inadequate ETL validation, or missing database constraints. We flag this explicitly to avoid the trap, noted by critics of RCA [16, 17], of a framework that explains everything and is therefore falsified by nothing. Establishing these causes would require triangulation the retrospective telemetry does not provide: commit history, controlled reproduction, and, critically, before/after recurrence.

Table 3
Causal hypotheses with evidence, alternatives, and confidence (author-assessed).

Deviation	Leading causal hypothesis	Supporting evidence	Alternative explanations	Confidence
Duplication / verification	Migration omitted canonical slot mapping and hashes	Code inspection; import logs; named pre-fix backups	Retry/batch artefacts; missing DB constraints; incomplete ETL validation	Moderate–High (mechanism)
Coverage deficiency	Slot taxonomy under-covers real document variety	Route/slot comparison; on-demand slot creation later added	Model uncertainty; expected by-design deferral	Moderate
Over-deferral at launch	0.92 gate mismatched to a bimodal confidence distribution	Confidence distribution; configuration-change record	Classifier miscalibration; distribution shift	Moderate (mechanism); Low that 0.70 is safe

4.6 The confidence threshold cannot be validated from telemetry

The classifier’s logged confidence (recorded on placement and verification actions) is markedly bimodal: of 6,685 scored actions, 2,453 sit at or above 0.98 and 4,206 below 0.70, with only 26 in the entire 0.70–0.98 band. The system launched with the auto-file threshold at 0.92 and it was later lowered to 0.70. It is tempting to read the bimodality as justifying the change, but the telemetry cannot support that inference, for two decisive reasons. First, there is no ground truth: the log does not record whether an auto-file was correct. Second, corrective actions do not carry the document or slot identifier, so an auto-file cannot be linked to any later correction, so the natural error proxy is unavailable (a linkage query returns zero matched slots). We therefore cannot compute accuracy conditional on confidence, a reliability curve, or a selective-risk/coverage curve, and cannot claim that 0.70 is a safe boundary. What we *can* say is conceptual and still useful: the threshold is a risk-acceptance decision (it fixes the residual automated error tolerated without human review), and it was changed without the evidence that would make that acceptance defensible. That gap, not the specific value, is the finding; supplying the missing calibration and risk-coverage evidence is a defined requirement for a future study.

4.7 The CAPA loop ran up to, but not including, effectiveness verification

The deployment ran an informal loop that our method makes explicit: detected deviations were contained, investigated, and addressed with corrective actions (de-duplication, re-slotting, re-filing, stale-pointer clearing) and preventive design changes (lowering the threshold; adding a slot-code inference step and a semantic document-to-slot index; a placeholder-detection guard; vision-based OCR with

hashed caching; tighter drafting prompts), each delivered as a versioned, dated deployment increment. Figure 3 renders this loop but marks its seventh stage (*verify effectiveness*) as required yet not implemented. This is the largest gap between the observed practice and genuine CAPA, and, as Section 3.2 established, it cannot be repaired retrospectively: because each corrective sweep ran on a single day with no subsequent operational window, there is no post-intervention period in which recurrence could be measured against a pre-defined criterion. The evidence thus demonstrates *RCA plus corrective intervention plus change logging* far more convincingly than *closed-loop CAPA*. Recognising precisely where the loop breaks is what lets us specify how to close it prospectively (Section 5).

5. Discussion

5.1 The contribution restated: an evidence specification for CAPA for AI

Read together, the findings convert a governance analogy into an evidence specification. Operational telemetry can support deviation detection, taxonomy construction, and, with care about the unit of analysis, prioritisation, and it can seed candidate causal hypotheses and document the corrective actions taken. It cannot, by itself, validate a decision threshold or demonstrate corrective-action effectiveness, and frequency counts can actively mislead prioritisation. A genuine “CAPA for AI” therefore requires four additions that the retrospective case lacked: incident-level denominators (Section 4.2); risk-weighted rather than count-weighted prioritisation (Section 4.3); causal-evidence provenance with triangulation (Section 4.5); and pre-registered effectiveness criteria evaluated over a real post-intervention window (Section 4.7). This makes the method both usable now and improvable: it tells the next deployment exactly what to instrument and pre-register so that the loop can actually close.

5.2 Deferral as designed oversight, and as an active-learning signal

The coverage-deficiency class illustrates a subtlety in meaningful human oversight [18] and connects to the selective-prediction literature [25, 26, 27]. A deferral is the safety mechanism working as intended; in aggregate, the same events map the boundary of the automation’s competence. The strongest forward direction the data suggests is to treat that boundary constructively: repeated deferrals on a document type are a machine-readable request for knowledge acquisition, which could drive taxonomy expansion, new slots, and additions to training and evaluation sets: an uncertainty-to-deferral-to-human-resolution-to-validated-change loop that is specific to AI rather than imported wholesale from manufacturing. Making that loop a *learning-to-defer* mechanism with consistent estimators and calibrated deferral rules [27, 37], rather than only a governance observation, would be a genuinely technical contribution beyond the CAPA framing.

5.3 The ethics of deploy-then-remediate in high-stakes settings

Honestly reported, the case is a story of deploying a system and remediating substantial defects afterwards. The pattern is common and not inherently unethical, but its obligations intensify with stakes, and immigration casework affects people's ability to live, work, and remain with their families. Deploy-then-remediate is defensible only when human oversight genuinely catches consequential errors before they reach the client, when remediation is disciplined rather than ad hoc, and when the residual risk accepted at each automation boundary is explicit and owned. The threshold decision in Section 4.6 is precisely such a risk-acceptance choice that was made without adequate evidence: not a claim that harm occurred, but an identification of where the governance was thinnest and where the discipline advocated here would bind.

5.4 What systems-safety would add

Our adoption of CAPA should not obscure the force of the systems-safety critique. STAMP/STPA would model the deployment as a control structure and ask which control actions and feedback loops were inadequate, resisting the search for a single root [31]; Safety-II would study how the automation usually succeeds, not only how it failed [32]. Both are complementary to, and in tension with, CAPA's more linear investigate-and-correct cadence. A mature governance practice for high-stakes AI would likely combine CAPA's auditable action-and-verification linkage with STPA's control-structure analysis and Safety-II's attention to normal operation. We present CAPA-for-AI as one operational pattern within that larger space, not as its resolution.

6. Limitations

This is a single-organisation, single-domain, retrospective case study, and its findings are analytic rather than statistically generalisable. The data is a compressed migration snapshot in which a single large backfill and single-day corrective sweeps dominate, precluding any longitudinal or before/after analysis. Deviation counts are counts of *remediated* deviations detected by the system's own logic and cannot capture failures the system did not detect; the true denominator of failures is unknown. Incident construction uses a case $\times$ type $\times$ day proxy in the absence of a stable per-document identifier across corrective actions; a document-level identifier would refine it. Causal statements are interpretive hypotheses subject to well-documented RCA limitations [16, 17] and to hindsight bias arising from the investigators' dual role as operators and analysts. The severity ratings in Table 2 are investigator judgments, not measurements, and the taxonomy has not been validated against a manually reviewed sample with inter-rater agreement, an obvious and important next step. Finally, the transferability of the pattern to other deployment classes (generative, conversational, recommender, or latent-outcome systems) is untested and should be evaluated rather than assumed.

7. Conclusion

We set out to test whether the quality-engineering disciplines of deviation management, RCA, CAPA, and change control provide a workable, cross-disciplinary method for governing AI automation, and whether an AI system's own telemetry can serve as the deviation record they require. The case study shows that

they can: on real operational data, the disciplines detect recurring failures, structure them into an auditable taxonomy, prioritise them, investigate them, and organise corrective and preventive action under change control. They also make the failures legible as an ethical matter, treating them as deviations for which an organisation is accountable rather than as background noise. The one stage the case could not complete from telemetry alone is formal effectiveness verification, and specifying precisely what that stage requires (incident-level denominators, risk-weighted prioritisation, causal-evidence provenance, and pre-registered effectiveness criteria evaluated prospectively) is itself part of the contribution: it turns an appealing analogy into a method a deployment can adopt and complete. Taking responsibility for what our systems do after we switch them on begins here, with the discipline to treat their failures as quality deviations and to close the loop on them.

Declarations

Competing interests.

A competing interest relevant to this work is disclosed on the title page and via the submission system, in accordance with the journal's double-anonymous policy. The analysis uses only aggregate, de-identified operational telemetry; no client-level data are reported. The dual role of the investigators as operators and analysts of the system is a potential source of interpretive bias, discussed in Sections 3.5 and 6.

Ethics approval.

Not applicable. The study is an operational audit of a live production system and involved no third-party human-subjects research; only aggregate, de-identified statistics are reported.

Use of generative AI.

Generative AI tools were used to assist with analysis scaffolding, figure preparation, and copy-editing. The authors reviewed and take full responsibility for the content.

Funding.

No funding was received for this work.

Author Contribution

Musarat Kabir-Chisty is the sole author. She conceived and designed the study, performed the data extraction and analysis, prepared all figures and tables, and wrote and approved the manuscript.

Data Availability

The data underlying this study are aggregate, de-identified operational telemetry from a live document-automation system used in confidential legal casework. Because the raw records comprise confidential client and personal data, they cannot be made publicly available and remain subject to client confidentiality and applicable data-protection law. No personal data are reported; all figures in the manuscript are aggregate counts and distributions. The derived, fully de-identified class counts underlying the tables and figures, together with the extraction logic (the class-to-action mapping and the incident de-duplication rule), are available from the corresponding author on reasonable request, subject to confidentiality and data-protection safeguards.

References

1. High-Level Expert Group on Artificial Intelligence: Ethics guidelines for trustworthy AI. European Commission, Brussels (2019)
2. National Institute of Standards and Technology: Artificial intelligence risk management framework (AI RMF 1.0), NIST AI 100-1. US Department of Commerce (2023). <https://doi.org/10.6028/NIST.AI.100-1>
3. International Organization for Standardization, International Electrotechnical Commission: ISO/IEC 42001:2023 Information technology – Artificial intelligence – Management system. ISO, Geneva (2023)
4. International Council for Harmonisation: ICH Q9(R1) Quality risk management. ICH, Geneva (2023)
5. International Council for Harmonisation: ICH Q10 Pharmaceutical quality system. ICH, Geneva (2008)
6. US Food and Drug Administration: Corrective and preventive action, historically codified at 21 CFR 820.100 (Quality System Regulation). Superseded, effective 2 February 2026, by the Quality Management System Regulation (QMSR), 89 FR 7496 (2 February 2024), which incorporates ISO 13485:2016 by reference. FDA, Silver Spring
7. Kabir-Chisty, M.: AI adoption as technology transfer: an integrative review and conceptual framework for governance, validation and knowledge preservation. *AI Ethics*. **6**, 434 (2026). <https://doi.org/10.1007/s43681-026-01296-9>
8. Selbst, A.D., Boyd, D., Friedler, S.A., Venkatasubramanian, S., Vertesi, J.: Fairness and abstraction in sociotechnical systems. In: *Proceedings of FAT* '19*, pp 59–68. ACM (2019). <https://doi.org/10.1145/3287560.3287598>
9. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., Dennison, D.: Hidden technical debt in machine learning systems. In: *Advances in Neural Information Processing Systems 28NIPS* pp 2503–2511 (2015) (2015)
10. Amershi, S., Weld, D., Vorvoreanu, M., et al.: Guidelines for human-AI interaction. In: *Proceedings of CHI '19*, paper 3. ACM (2019). <https://doi.org/10.1145/3290605.3300233>

11. Raji, I.D., Smart, A., White, R.N., Mitchell, M., Gebru, T., Hutchinson, B., Smith-Loud, J., Theron, D., Barnes, P.: Closing the AI accountability gap: defining an end-to-end framework for internal algorithmic auditing. In: Proceedings of FAT* '20, pp 33–44. ACM (2020).
<https://doi.org/10.1145/3351095.3372873>
12. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., Gebru, T.: Model cards for model reporting. In: Proceedings of FAT* '19, pp 220–229. ACM (2019).
<https://doi.org/10.1145/3287560.3287596>
13. International Organization for Standardization: ISO 13485:2016 Medical devices – Quality management systems – Requirements for regulatory purposes. ISO, Geneva (2016)
14. Rooney, J.J., Vanden Heuvel, L.N.: Root cause analysis for beginners. *Qual. Prog.* **37**(7), 45–53 (2004)
15. International Organization for Standardization: ISO 9001:2015 Quality management systems – Requirements. ISO, Geneva (2015)
16. Peerally, M.F., Carr, S., Waring, J., Dixon-Woods, M.: The problem with root cause analysis. *BMJ Qual. Saf.* **26**(5), 417–422 (2017). <https://doi.org/10.1136/bmjqs-2016-005511>
17. Card, A.J.: The problem with '5 whys'. *BMJ Qual. Saf.* **26**(8), 671–677 (2017).
<https://doi.org/10.1136/bmjqs-2016-005849>
18. de Santoni, F., van den Hoven, J.: Meaningful human control over autonomous systems: a philosophical account. *Front. Robot AI.* **5**, 15 (2018). <https://doi.org/10.3389/frobt.2018.00015>
19. Ishikawa, K.: Guide to quality control, 2nd edn. Asian Productivity Organization, Tokyo (1986)
20. Reason, J.: Human error. Cambridge University Press, Cambridge (1990)
21. Lee, J.D., See, K.A.: Trust in automation: designing for appropriate reliance. *Hum. Factors.* **46**(1), 50–80 (2004). https://doi.org/10.1518/hfes.46.1.50_30392
22. Parasuraman, R., Manzey, D.H.: Complacency and bias in human use of automation: an attentional integration. *Hum. Factors.* **52**(3), 381–410 (2010). <https://doi.org/10.1177/0018720810376055>
23. International Organization for Standardization, International Electrotechnical Commission: ISO/IEC 23894:2023 Information technology – Artificial intelligence – Guidance on risk management. ISO, Geneva (2023)
24. Ulnicane, I., Knight, W., Leach, T., Stahl, B.C., Wanjiku, W.G.: Governance of artificial intelligence: emerging international trends and policy frames. In: Tinnirello, M. (ed.) *The global politics of artificial intelligence*, pp. 29–55. CRC, Boca Raton (2022)
25. Geifman, Y., El-Yaniv, R.: Selective classification for deep neural networks. In: *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp 4878–4887 (2017)
26. Madras, D., Pitassi, T., Zemel, R.: Predict responsibly: improving fairness and accuracy by learning to defer. In: *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)* arXiv:1711.06664 (2018)

27. Mozannar, H., Sontag, D.: Consistent estimators for learning to defer to an expert. In: Proceedings of the 37th International Conference on Machine Learning (ICML 2020), PMLR 119, pp 7076–7087 (2020)
28. Rabanser, S., Günnemann, S., Lipton, Z.C.: Failing loudly: an empirical study of methods for detecting dataset shift. In: Advances in Neural Information Processing Systems 32NeurIPS (2019) (2019)
29. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in AI safety. (2016). arXiv:1606.06565
30. Varshney, K.R., Alemzadeh, H.: On the safety of machine learning: cyber-physical systems, decision sciences, and data products. *Big Data*. **5**(3), 246–255 (2017). <https://doi.org/10.1089/big.2016.0051>
31. Leveson, N.G.: Engineering a safer world: systems thinking applied to safety. MIT Press, Cambridge (2011)
32. Hollnagel, E.: Safety-I and Safety-II: the past and future of safety management. Ashgate, Farnham (2014)
33. European Parliament and Council: Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official J. Eur. Union (2024)
34. Autio, C., Schwartz, R., Dunietz, J., Jain, S., Stanley, M., Tabassi, E., Hall, P., Roberts, K.: Artificial intelligence risk management framework: generative artificial intelligence profile, NIST AI 600-1. National Institute of Standards and Technology (2024). <https://doi.org/10.6028/NIST.AI.600-1>
35. McGregor, S.: Preventing repeated real world AI failures by cataloging incidents: the AI Incident Database. In: Proceedings of the AAAI Conference on Artificial Intelligence 35(17), 15458–15463 (2021)
36. Paleyes, A., Urma, R.-G., Lawrence, N.D.: Challenges in deploying machine learning: a survey of case studies. *ACM Comput. Surv.* **55**(6), 114 (2022). <https://doi.org/10.1145/3533378>
37. Verma, R., Nalisnick, E.: Calibrated learning to defer with one-vs-all classifiers. In: Proceedings of the 39th International Conference on Machine Learning (ICML 2022), PMLR 162, pp 22184–22202 (2022)

Figures

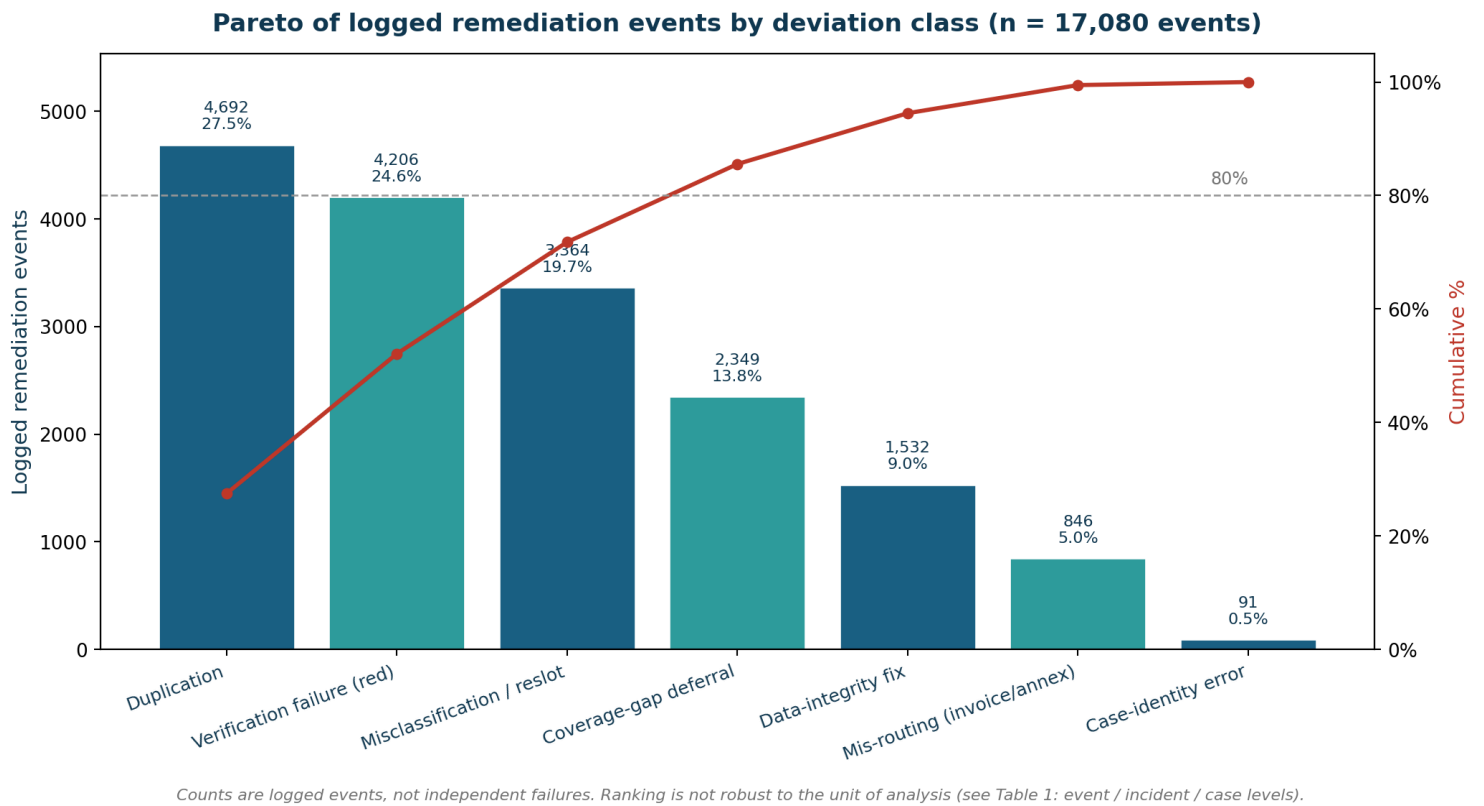


Figure 1

Pareto of logged remediation events by deviation class. Bars are logged events, not independent failures; the ranking is not robust to the unit of analysis (Table 1).

Candidate contributing factors identified in retrospective RCA (investigator-generated)

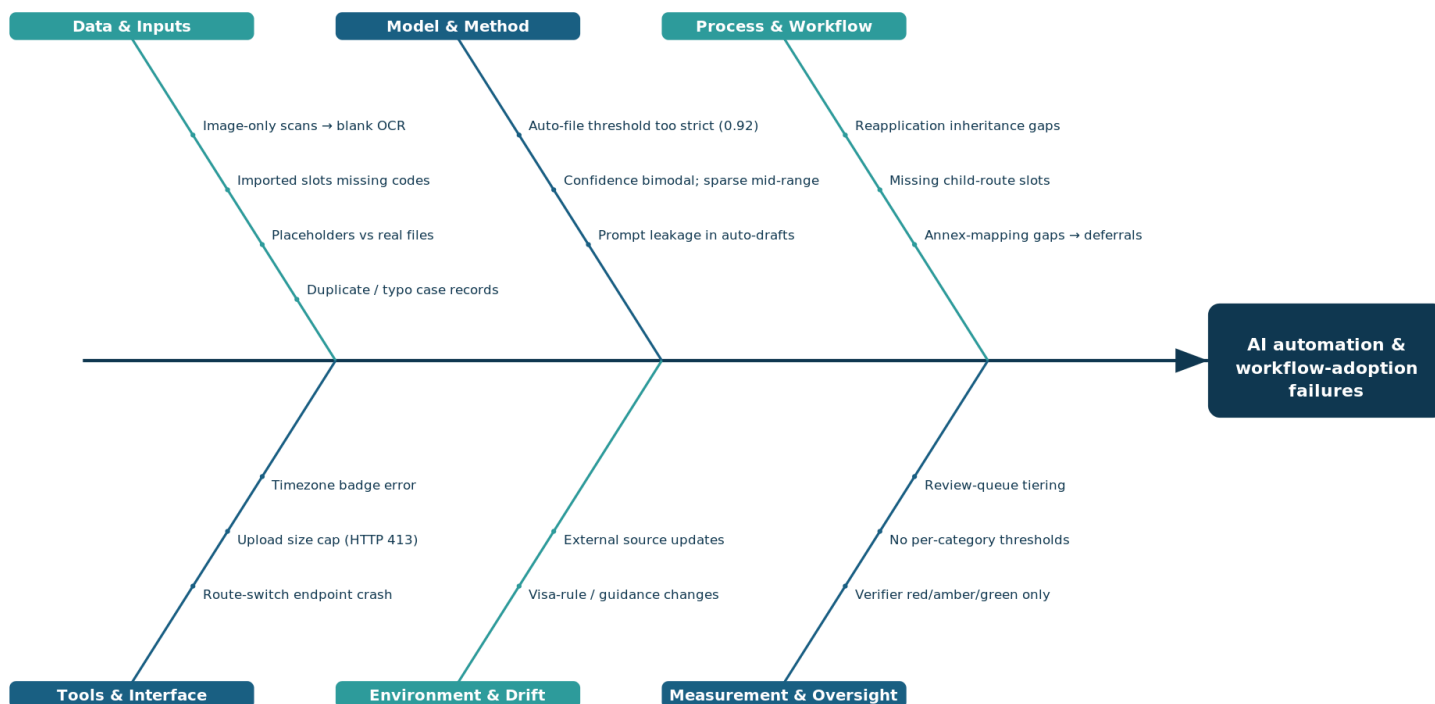


Figure 2

Candidate contributing factors identified in retrospective, investigator-generated RCA. The diagram structures hypotheses; it does not establish causation.

A CAPA and change-control loop for AI automation (dashed = required but not implemented in the case)

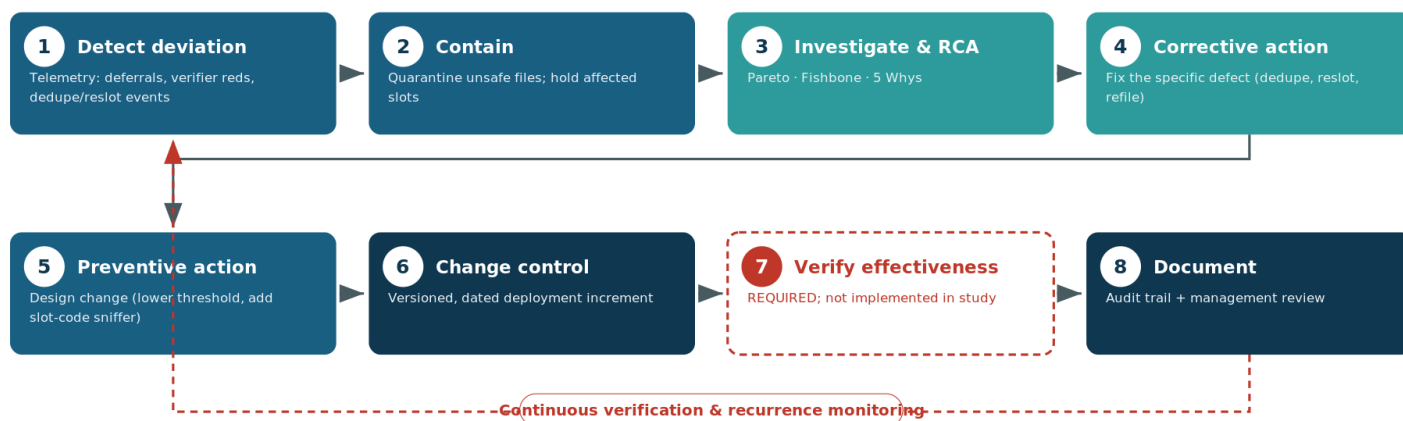


Figure 3

A CAPA and change-control loop for AI automation. Solid stages were observed in the case; the dashed stage (verify effectiveness) is required but was not implemented and is not reconstructable from the data.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Appendix.docx](#)