

AI-LEx: An AI-Driven Interoperability Architecture for Laboratory Exception Management, Patient Safety, and Public Health

Sai Krishna Puli^{1*}, Priya Bandhavi Pichika²

¹Data Architecture, Independent Researcher, Georgia, USA

²HealthCare Data Interoperability, Independent Researcher, Georgia, USA

Email: *naprime2748@gmail.com, priyapichika@gmail.com

How to cite this paper: Puli, S.K. and Pichika, P.B. (2026) AI-LEx: An AI-Driven Interoperability Architecture for Laboratory Exception Management, Patient Safety, and Public Health. *International Journal of Intelligence Science*, 16, 273-287. <https://doi.org/10.4236/ijis.2026.164015>

Received: September 15, 2026

Accepted: October 20, 2026

Published: October 23, 2026

Copyright © 2026 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0). <http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

AI-LEx introduces a risk-aware architecture that transforms laboratory interoperability exceptions from technical failures into clinically and publicly meaningful risk signals, by combining deterministic validation, machine-learning anomaly detection, semantic knowledge, and human-governed resolution. A missing specimen source or an unmapped code on a reportable pathogen looks, now it occurs, like a routine interface problem; left unresolved, it can instead become a missed diagnosis or an invisible gap in disease surveillance. Recent empirical work tracing real laboratory transmissions across health information exchanges documents exactly this pattern, and the FDA's decade-long SHIELD initiative and a 2024 federal data brief both identify laboratory-data harmonization as an unresolved patient-safety and surveillance concern. AI-LEx addresses it with a three-stage architecture: Detection combines deterministic validation with a calibrated machine-learning anomaly classifier; Interpretation grounds each detected irregularity in semantic knowledge via a knowledge graph and retrieval-augmented generation; and Risk & Resolution computes a Risk Score—Consequence × Propagation × Urgency—kept strictly independent of the AI's own confidence, and acts on the result only through human-governed resolution: an explicit decision matrix rather than a blended score. A synthetic, illustrative evaluation comparing this hybrid architecture against single-method baselines shows an F1 of 0.983 against 0.278 - 0.833, showing that an OR-combined detector recovers signal that no single channel could—under a generator deliberately constructed around independent channels, not as a general claim about hybrid detection on operational laboratory data. We conclude with the architecture's implications, its limitations, and the empirical work required before clinical deployment.

Keywords

Laboratory Information Systems, Semantic Interoperability, HL7 FHIR, Artificial Intelligence, Exception Management, Risk Scoring, Public Health Surveillance

1. Introduction

A physician orders a basic metabolic panel. Behind that single click, a laboratory information system, interface middleware, an electronic health record, and a web of terminology mappings must agree, code for code and unit for unit, on what “potassium, serum” means. Most of the time this agreement is invisible. When it is not, the failure is filed under a deliberately bland word—an “exception”—that can range from a transposed digit to a positive culture for a reportable pathogen that never gets flagged, because a local code was never mapped to a standard concept.

Most exceptions never matter. A duplicate result a technologist discards, a message that gets retried a second later and quietly succeeds—these are the background noise of running a laboratory, not a research problem. This paper cares about a narrower and more consequential slice: laboratory data exceptions that can turn into patient-safety or public-health events. What separates them from the background noise isn’t how alarming they look the moment they’re detected, but their propagation potential, meaning whether an exception left unresolved can reach a clinician’s decision or a public-health case count before anyone notices. A 2024 case report tracing real laboratory transmissions across a health information exchange found exactly this failure mode in practice [1]: specimen source missing, units and reference ranges inconsistent, coding discordant, HL7 versions clashing. Separately, the FDA’s Systemic Harmonization and Interoperability Enhancement for Laboratory Data (SHIELD) initiative [2] and a 2024 federal data brief [3] each identify laboratory-data harmonization as unresolved and consequential for patient safety and surveillance alike.

The central move in this paper is to stop treating a laboratory interoperability exception as a bare technical failure and start treating it as a risk signal. The AI does not predict clinical abnormalities or diagnose the patient; it recognizes irregularities in the data and workflow surrounding a result—an unmapped code, a malformed message, an identity mismatch—and assigns each one a risk score describing what would happen if it propagated into clinical care or public-health reporting. AI-LEx performs this transformation by combining four capabilities: deterministic validation and machine-learning anomaly detection to catch the irregularity, semantic knowledge to explain it, and human-governed resolution to act on it safely. Cutting any one of these four leaves the transformation incomplete—validation alone cannot explain what it catches, semantic knowledge alone cannot decide when to act, and neither is safe to act on without a human able to

intervene—and this is the paper’s organizing contribution.

Can an AI-driven, interoperable laboratory exception-management architecture detect, classify, explain, prioritize, and resolve laboratory data exceptions across healthcare and public-health systems before they create patient-safety or public-health risk?

Section 2 reviews related work in three areas. Section 3 states the resulting research gap and four research questions. Section 4 presents the AI-LEx architecture across four diagrams. Section 5 describes a synthetic experimental design, and Section 6 reports its results. Sections 7 and 8 discuss patient-safety and public-health implications and the architecture’s limitations, and Section 9 concludes.

2. Related Work

2.1. Laboratory Interoperability

Laboratory data moves through a layered stack of standards. HL7 v2 messaging is still the workhorse of most laboratory interfaces, though FHIR [4] is increasingly positioned to take over as its API-based successor. Underneath the messaging layer, LOINC identifies what test was performed [5], SNOMED CT captures results and organisms [6], and UCUM standardizes units of measure. A proof-of-concept study combining FHIR with SNOMED CT [7] made an important point along the way: getting the message shape right (structural interoperability) and getting the meaning right (semantic interoperability) are genuinely separate engineering problems and solving one does not solve the other. That distinction shows up constantly in practice, because local test codes tend to multiply faster than any institution’s mapping tables can keep up with them. One 2024 case report [1] followed nine laboratory tests as they moved between two institutions over a standard point-to-point health information exchange, and found the kind of quiet information loss that rarely makes it into a bug ticket: specimen source went unreported, the analytical technique behind a result was never communicated, reference ranges didn’t match across the two systems, LOINC coding was inconsistent, and the coexistence of multiple HL7 versions added yet another layer of friction. The authors pointed toward the FDA’s SHIELD initiative [2]—a harmonization effort that has been running since 2015 across more than seventy participating organizations—as the kind of coordinated response a problem at this scale requires.

2.2. AI in Laboratory Quality Management

AI has made its most visible inroads in the pre-analytical phase, which makes sense given that this is where the largest share of laboratory errors is already known to originate—precisely the phases the ISO 15189 laboratory-quality standard is built to govern [8], a pattern documented since at least the mid-2000s [9] and confirmed more recently by site-specific prevalence studies in Iraqi Kurdistan [10] and Morocco [11]. Image-based clot detection, wrong-blood-in-tube identification, and automated hemolysis/icterus/lipemia assessment have all reported

strong published performance [12] [13], and artificial neural networks have found a home in autoverification, the process of deciding whether a result can be released without a human looking at it first [14] [15]. What's still largely missing is anything connecting these point solutions to one another. A clot-detection model and a delta-check model can both fire on the same specimen at the same time, and neither one can explain its finding in terms the other, or a downstream reporting system, would understand.

2.3. Public Health Reporting

Reportable conditions flow to public health authorities through NNDSS [16]. A federal data brief published in September 2024 [3] makes the same point from the surveillance side that the case report above [1] made from the clinical side: interoperable laboratory data matter both for bedside decisions and for population-level monitoring, and the brief is refreshingly candid that getting there is still complicated and unfinished business. Newer FHIR-based electronic case reporting is gradually replacing the older batch HL7 v2 file transfers, but it doesn't remove the underlying vulnerability—if a reportable result's local code was never mapped correctly further upstream, the case can simply vanish from public-health review without anyone noticing.

3. Research Gap and Questions

Point-specific AI models for pre-analytical error detection exist and perform well within narrow scope. Semantic interoperability standards exist and are mature, if unevenly adopted. Public-health reporting channels exist and are being modernized toward FHIR. What does not widely exist is an architecture that unifies detection, semantic grounding, risk-aware prioritization, and human governance into one auditable pipeline that treats an exception's full lifecycle as a single continuous problem rather than a series of handoffs between separate systems. This gap motivates four research questions:

- RQ1—Can a hybrid architecture combining deterministic rules, calibrated ML, and knowledge-grounded generation detect and classify laboratory data/workflow exceptions more effectively than any single method alone?
- RQ2—Can risk and AI confidence be computed as independent signals and combined through an explicit decision matrix to reduce false positives without sacrificing recall on high-risk cases?
- RQ3—What architectural mechanisms keep such a system auditable and appropriately human-governed rather than autonomously making irreversible decisions?
- RQ4—Can this architecture's design plausibly address the categories of interoperability failure already documented empirically in real laboratory-to-EHR data transmissions?

Section 4 proposes an architecture addressing all three of RQ1 - RQ3, but only RQ1 receives empirical evaluation in this paper, in Sections 5 and 6; RQ2 and RQ3

are addressed at the level of architectural design and remain to be tested empirically. RQ4 is addressed by direct reference to the empirical failure modes reviewed in Section 2.1.

4. AI-LEx Architecture

Each of AI-LEx's four defining capabilities has a home in the sections below: deterministic validation and machine-learning anomaly detection are realized in Stage A, and semantic knowledge in Stage B—both within Section 4.2—and human-governed resolution in Stage C (Section 4.3), where a risk score is finally attached to what Stage A and B detected and explained. AI-LEx is presented across four diagrams rather than one increasingly overloaded picture: an overall pipeline view, the AI detection and reasoning mechanism, the risk and resolution mechanism that performs this paper's central transformation, and the end-to-end workflow that ties them together.

4.1. Overall Architecture

Figure 1 traces one pipeline from source to outcome: laboratory data sources (LIS, analyzers, EHR, point-of-care devices, reference and public-health labs) enter through an interoperability layer (HL7 v2, FHIR, ELR, eCR, HIE), pass through semantic harmonization against LOINC, SNOMED CT, UCUM, and a master patient index, and are evaluated by the AI exception engine before classification, risk scoring, and human-governed resolution. The architecture is designed to sit alongside existing laboratory and health-IT systems rather than replace them.

4.2. AI Detection and Reasoning

Figure 2 addresses RQ1. Stage A, Detection, runs a deterministic rules engine and a calibrated ML classifier side by side—rules catch what is already known, the classifier catches what is merely like something known, and neither substitutes for the other. Stage B, Interpretation, is triggered when Stage A's output needs grounding: a knowledge graph supplies ontology and causal-link context, and a retrieval-augmented language model synthesizes an explanation from similar past cases, constrained to cite retrieved evidence rather than reason freely from training data alone. Grounding generation in a structured knowledge graph rather than letting a model reason unconstrained is a design choice with growing support in clinical AI research, where knowledge-graph-based reasoning has been shown to improve both interpretability and reliability for clinical decision support [17]–[19].

4.3. Risk and Resolution

Figure 3 addresses RQ2 as a proposed design and is the architecture's central methodological discipline; it is not empirically evaluated in this paper, a scope Section 5 states explicitly. We define a Risk Score independent of, and never adjusted by, the AI's confidence in its own classification—how certain a model is that it has classified an exception correctly is not evidence of how dangerous that

Figure 1 — Overall Architecture

One pipeline, top to bottom, from source system to resolved outcome

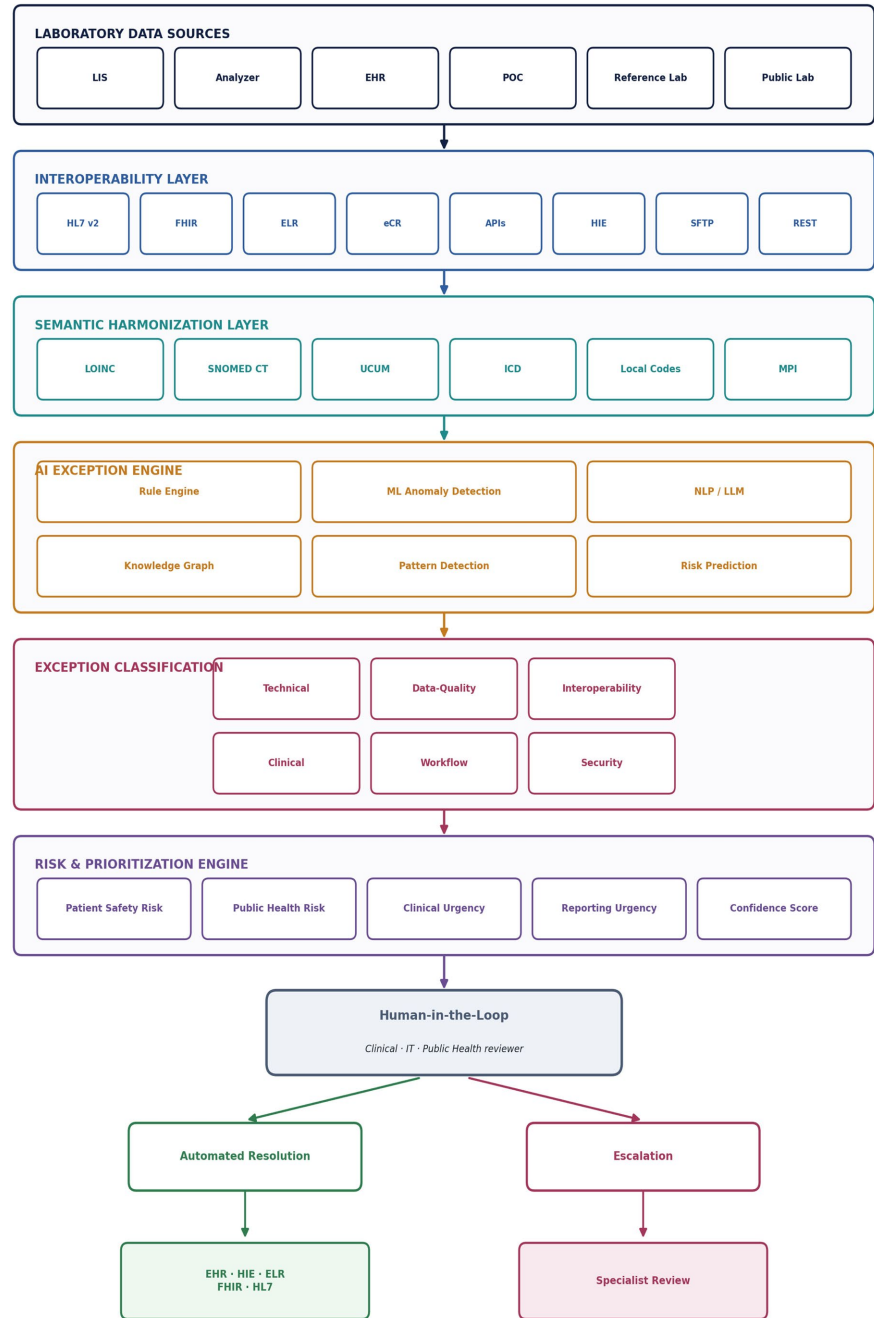


Figure 1. Overall architecture: one pipeline from source system to resolved outcome.

exception actually is, and a system that lets high confidence quietly lower a risk figure can be badly wrong about the cases it is most sure about.

$$Risk = Consequence \times Propagation \times Urgency$$

Consequence estimates clinical and public-health harm if unresolved; Propagation estimates how far the underlying data problem could spread before detection; Urgency estimates the time remaining before Consequence becomes irreversible.

AI Confidence is computed separately, as a calibrated estimate of certainty in the classification only, and the two scores meet solely at the decision matrix in **Figure 3**: low risk with high confidence auto-resolves; high risk, even with high confidence, receives quick human validation rather than silent auto-resolution; low confidence at high risk escalates immediately; and any critical-risk case reaches full human review regardless of confidence. This is where the paper’s central transformation is performed: a detected, explained irregularity becomes a risk signal only once Consequence, Propagation, and Urgency are attached to it, and that risk signal is acted on only through human-governed resolution—never by the AI unilaterally, regardless of how confident it is.

Figure 2 — AI Detection & Reasoning

Two stages: catching what is already known, then grounding what is merely similar

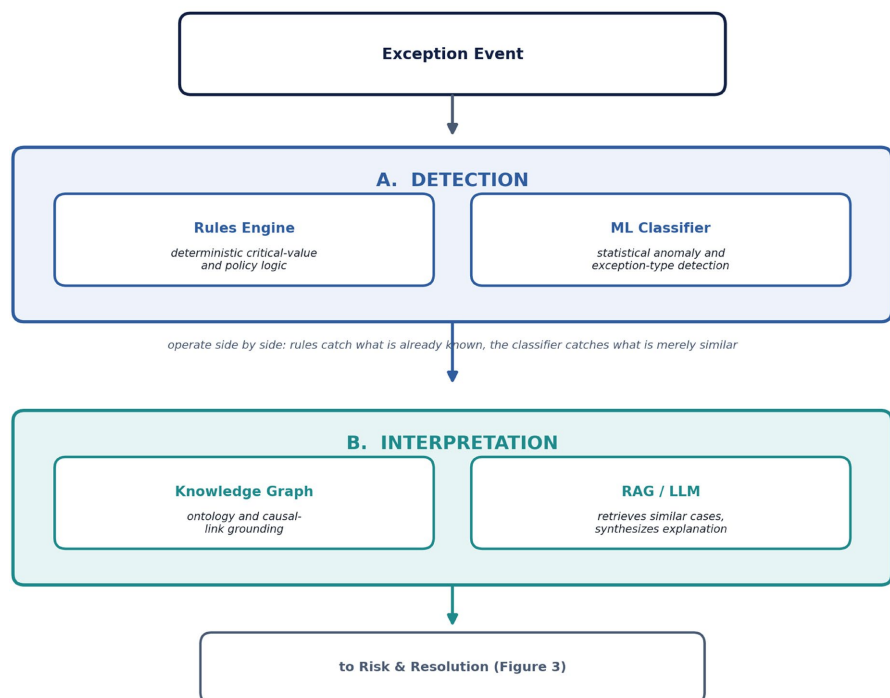


Figure 2. AI detection and reasoning: two stages, each an independently auditable capability.

4.4. End-to-End Workflow

Figure 4 addresses RQ3. A single shared twelve-step pipeline—detection, classification, prioritization, investigation, AI recommendation, confidence assessment, human review, escalation, resolution, notification, audit, and feedback—carries every exception, with type assigned once at classification across six categories (technical, data-quality, interoperability, clinical, workflow, security) that steer routing without altering the pipeline itself. An auto-resolution shortcut and an escalation path give the decision matrix concrete routes to act on, and every step, automated or human, is written to an immutable audit log, keeping the human reviewer—not the model—the final authority on any case the matrix does not

clear for auto-resolution.

5. Experimental Design

No real-world labeled corpus of laboratory exceptions was available for this paper, and we make no claim to clinical validation. Instead, we report a synthetic, illustrative evaluation designed to test a narrow, falsifiable version of RQ1: whether combining independent detection channels outperforms any single channel, on data constructed so that no single channel has access to all the signal that determines ground truth. To be explicit about scope: Sections 5 and 6 evaluate RQ1 only. RQ2’s decision mechanism (Section 4.3) is a proposed design, not an evaluated one—doing so would require risk-category ground truth, confidence-threshold behavior, and outcomes for each of the five decision-matrix actions, none of which this binary-detection experiment constructs. We limit our empirical claims to RQ1 accordingly and return to this scope explicitly in Section 6.

Figure 3 — Risk & Resolution

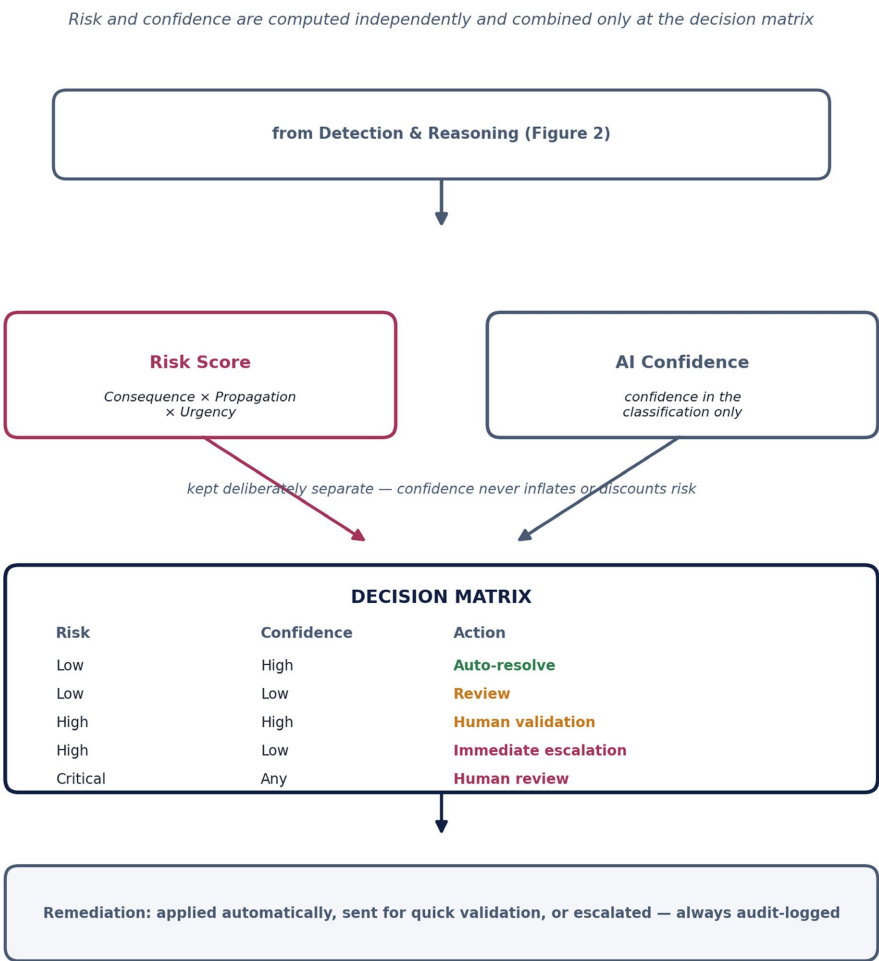


Figure 3. Risk and resolution: risk and confidence computed independently, combined only at the decision matrix.

Figure 4 — Exception Management Workflow

A single shared pipeline; exception TYPE is determined once, at classification, and steers routing downstream

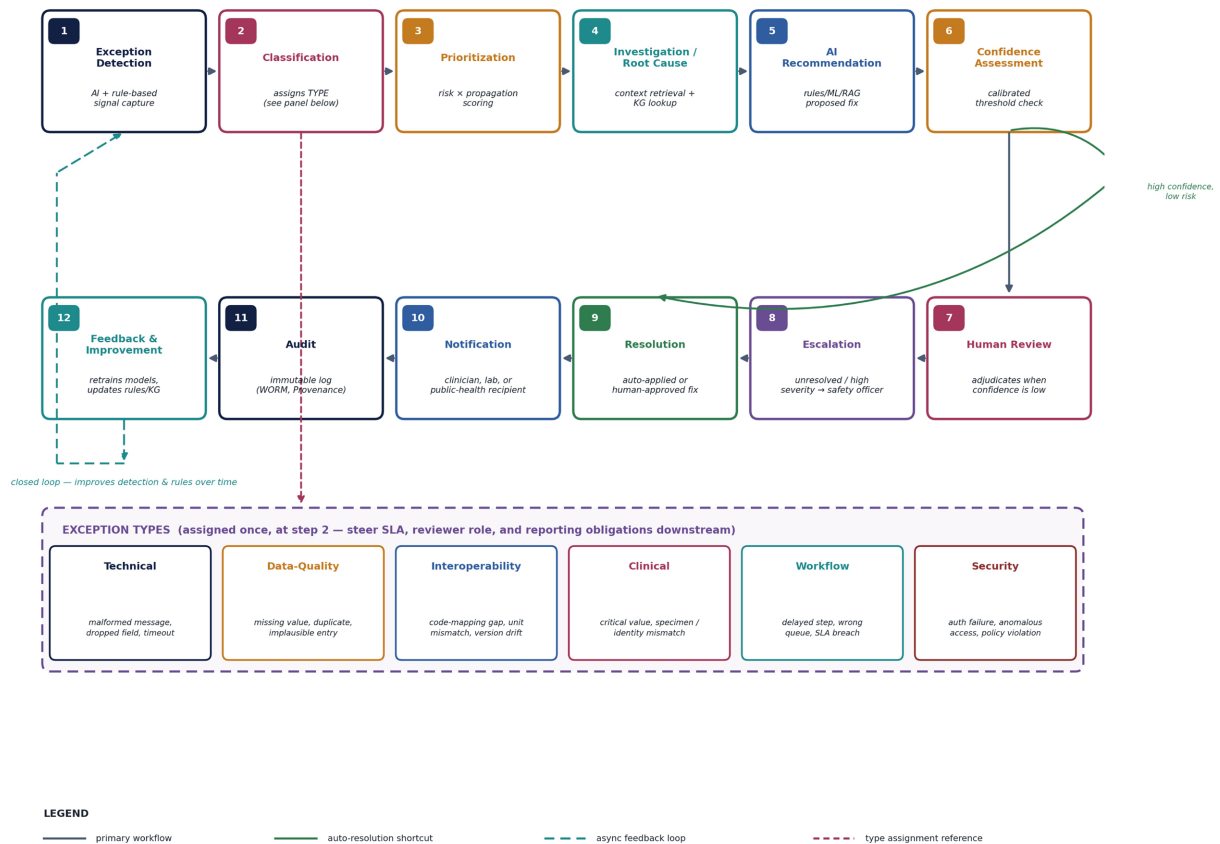


Figure 4. End-to-end workflow: a shared twelve-step pipeline with an auto-resolution shortcut, an escalation path, and six exception types assigned once at classification.

Before presenting the synthetic labels, it is worth being precise about what they represent. Following the propagation-potential criterion introduced in Section 1, we define a genuine actionable exception as one that, left unresolved, would either alter or delay a clinical decision or cause a case to fail to surface for public-health reporting when it should have. This is deliberately a criterion about downstream consequence, not about how alarming or unusual the underlying anomaly looks now it is flagged. A routine technical exception—a duplicate message that resolves itself on retry, a cosmetic formatting mismatch with no bearing on interpretation—fails this criterion even if it trips a system-level alert, because nothing downstream would change if it went unaddressed. An actionable interoperability event, by contrast, is one where that downstream change is real: a code that stays unmapped, a specimen source that stays missing, a result that reaches a clinician or a surveillance system already wrong. Section 5.1 operationalizes exactly this distinction rather than a looser notion of “anomalous”.

5.1. Synthetic Dataset

We generated $N = 3,000$ synthetic records with a fixed master random seed (NumPy

Generator, seed 42), split 70/30 into training (2,100) and test (900) partitions via stratified sampling on the label (scikit-learn `train_test_split`, `random_state = 7`). Every distribution and threshold below is exact and sufficient, together with the seed, to reproduce the dataset and every result in Section 6 bit-for-bit.

Structured deterministic channel (favors a rules approach)—four independent Bernoulli draws: `code_mapped` is False with probability 0.35; `units_consistent` is False with probability 0.20; `specimen_source_present` is False with probability 0.25; `organism_reportable` is True with probability 0.15.

Structured statistical channel (favors an ML classifier)—three continuous, independently drawn features: `message_malformed_score` $\sim \text{Beta}(\alpha = 2, \beta = 5)$ (mean ≈ 0.286); `historical_deviation_score` $\sim \text{Normal}(\mu = 0, \sigma = 1)$; `field_completeness` $\sim \text{Beta}(\alpha = 5, \beta = 2)$ (mean ≈ 0.714).

Free-text channel (favors a retrieval-style method)—each record draws a latent-risk value from Uniform (0, 1), then selects one of six fixed templates from a categorical distribution conditioned on that draw: {`clotted_redraw`: 0.28, `wrong_patient`: 0.22, `routine_admin`: 0.03, `duplicate`: 0.05, `reportable_organism_note`: 0.32, `generic`: 0.10} when the latent value exceeds 0.6, and {0.05, 0.03, 0.35, 0.15, 0.02, 0.40} in the same template order otherwise. The `clotted_redraw`, `wrong_patient`, and `reportable_organism_note` templates are text-channel-positive; the other three are not.

Latent-risk thresholds defining ground truth—a record is exception-positive on the rules channel if (`code_mapped` is False AND `organism_reportable` is True) OR (`units_consistent` is False AND `specimen_source_present` is False); positive on the ML channel if `message_malformed_score` > 0.55 OR `historical_deviation_score` > 1.3 OR `field_completeness` < 0.35 ; positive on the text channel if its template is text-channel-positive as defined above.

Label-noise implementation—independent of all three channels, each record is flipped to exception-positive with probability 0.04 (an independent Bernoulli (0.04) draw per record), representing irreducible labeling uncertainty. Ground truth is the logical OR of the three channel signals and the noise flip.

Resulting class allocation—not independently imposed, but an emergent property of the generator above: 57.1% of the full dataset (and 57.1% of the 900-record test partition, by construction of the stratified split) is exception-positive.

5.2. Probability Calibration

The random-forest classifier used in the ML-only and hybrid approaches (Section 5.3) produces raw class probabilities that are not, by default, calibrated—tree-ensemble vote fractions are a well-documented source of over- or under-confident probability estimates. We calibrate them explicitly rather than assume calibration: the base `RandomForestClassifier` (200 trees, max depth 6, `random_state = 1`) is wrapped in isotonic-regression calibration fit via 5-fold stratified cross-validation (scikit-learn `CalibratedClassifierCV`, `cv` seed 99) internal to the 2,100-record training partition. The 900-record test partition is never used for fitting or calibration,

only for the final metrics reported in Section 6. Calibration quality is measured by Brier score on the held-out test partition: 0.187 for the uncalibrated model versus 0.182 calibrated—a modest, honestly reported improvement, not a dramatic one. All ML-only and hybrid results in Section 6 use the calibrated probabilities. The LLM/RAG-proxy’s logistic-regression probabilities are not separately calibrated, since they are already the model’s native training objective rather than a post-hoc vote fraction; we do not claim they are calibrated in the same sense.

5.3. Exception Classes and Baselines

Records are drawn across the same six exception types used in **Figure 4** (technical, data-quality, interoperability, clinical, workflow, security). Four approaches were evaluated on the identical held-out test set, each with an explicit decision threshold:

- Rules-only—the deterministic channel alone: fixed logical conditions on structured fields (Section 5.1), no learned threshold, no learning.
- ML-only—the calibrated random-forest classifier described in Section 5.2, flagged positive when its calibrated probability exceeds 0.5.
- LLM/RAG-proxy—a bag-of-words logistic-regression classifier trained on free text alone, flagged positive when its probability exceeds 0.5; standing in for a retrieval/LLM-style approach since no live LLM inference was run in this synthetic setting, a substitution we state plainly as a limitation in Section 8.
- AI-LEx (hybrid)—a case is flagged if any of the three channels above crosses its own threshold (the rules channel’s fixed logic, or a 0.5 probability cutoff for ML or text), mirroring the architecture’s principle that a strong signal from one channel is sufficient to warrant review without requiring cross-channel agreement.

5.4. Metrics

We report precision, recall, and F1 for the binary decision “is this a genuine actionable exception”, plus the false-positive count on the 900-record test set, chosen because Section 7 identifies alert fatigue—not missed detection alone—as a primary mechanism by which an exception-management system fails in practice.

6. Results

Table 1 reports precision, recall, F1, and false positives for all four approaches on the synthetic held-out test set ($n = 900$; base rate of genuine exceptions 57.1%).

Table 1. Synthetic, illustrative evaluation results—not a clinical validation. AI-LEx (hybrid) row highlighted.

Approach	Precision	Recall	F1	False positives
Rules-only	100.0%	16.1%	27.8%	0
ML-only	97.5%	45.5%	62.1%	6
LLM/RAG-proxy	100.0%	71.4%	83.3%	0
AI-LEx (hybrid)	98.8%	97.9%	98.3%	6

Rules-only achieved perfect precision (100.0%) but the lowest recall (16.1%, $F1 = 0.278$): deterministic conditions caught only the subset of exceptions their fixed logic was written for, exactly the failure mode Section 2.1's empirical evidence describes—correct when it fires, silent otherwise. ML-only, using the calibrated probabilities described in Section 5.2, reached moderate performance ($F1 = 0.621$, precision 97.5%, recall 45.5%) and produced 6 of the evaluation's false positives, consistent with a statistical classifier occasionally flagging borderline structured patterns that free text or hard rules would not. The LLM/RAG-proxy performed better than single structured method ($F1 = 0.833$) with zero false positives, reflecting that this synthetic dataset's text channel was a comparatively clean signal—a property of the synthetic generator, not a general claim about retrieval-based methods. AI-LEx (hybrid) achieved the highest recall (97.9%) and $F1$ (0.983) of all four approaches, inheriting ML-only's 6 false positives unchanged, since the hybrid decision rule flags a case if any channel does.

This pattern is consistent with RQ1 as posed: because the dataset's ground truth was constructed so that no single channel sees all the signal, no single-channel method could recover more than a fraction of true exceptions, while an OR-combined detector drawing on all three channels recovered nearly all of them. The false-positive count carried over unchanged from ML-only to the hybrid is itself informative: it shows that combining channels here improved recall without a corresponding recall-driven increase in false positives—the hybrid rule did not add new false positives beyond what its noisiest single channel already produced. We are deliberately narrow about what this shows: an OR-combined detector performs well under a generator designed around independent channel signals. It does not demonstrate the general logical necessity of hybrid detection in operational laboratory data, where channels may be correlated, noisier, or informative in ways this synthetic construction does not capture. We return to this distinction, and to what would be required to test it on real data, in Section 8.

7. Patient Safety and Public Health Implications

For patient safety, the architecture's contribution is not that it catches more exceptions than existing point solutions—several detection techniques already exist individually—but that every exception it surfaces already carries a clinically meaningful risk signal rather than a bare technical flag and is resolved automatically only when that signal says it is safe to do so. Alert fatigue is a well-documented hazard in clinical informatics, and a system that surfaces only the exceptions genuinely requiring human judgment, each with a specific, ontology-grounded rationale and an explicit risk score attached, is more likely to receive the attention it needs than one that treats every anomaly as equally urgent. Reducing time-to-resolution matters most exactly where speed is clinically important: critical values, unstable patients, and reportable pathogens.

For public health, the architecture targets a specific and under-appreciated cause of underreporting: cases that are clinically identified but never surface for

public-health review because of a semantic mapping gap rather than any failure of clinical judgment, precisely the failure mode documented in the 2024 case report reviewed in Section 2.1. Because AI-LEx assigns a public-health-impact term inside the Risk Score itself, a mapping gap on a reportable organism does not merely fail silently—it is surfaced as a publicly meaningful risk signal before it reaches the reporting layer, reducing the number of cases that fail to reach surveillance systems and improving reporting speed, which matters disproportionately during an active outbreak.

8. Limitations and Future Work

Three limitations qualify every claim in this paper. First, and most importantly, Sections 5 and 6 report a synthetic, illustrative evaluation of RQ1 only, not a clinical validation and not a test of RQ2 or RQ3: the dataset was constructed by the authors specifically so that no single channel could recover the full label, which shows that an OR-combined detector outperforms any single channel under this generator but cannot estimate real-world precision, recall, or false-positive rates on genuine laboratory data, and says nothing yet about whether the proposed risk/confidence decision matrix (RQ2) performs as intended. Second, the “LLM/RAG-proxy” baseline is a bag-of-words classifier standing in for a retrieval/LLM-style method; no live large-language-model inference was run, and a real RAG pipeline’s performance on this task remains untested. Third, a newly deployed system’s exception knowledge graph starts empty, and whether federating generalized error patterns across institutions—without pooling identifiable data—can close this cold-start gap without eroding the specificity that makes the knowledge graph useful is an open empirical question this paper does not resolve.

The clear next step is validating this architecture against a real, labeled corpus of laboratory exceptions, ideally spanning multiple institutions, replacing the text-classifier proxy with an actual retrieval-augmented LLM pipeline—an approach recent work has begun applying directly to automated construction of clinical knowledge graphs [20]—and prospectively measuring alert-fatigue outcomes among reviewing clinicians and laboratory staff rather than relying on the synthetic false-positive count reported here. Evaluating RQ2 and RQ3 empirically is a separate and necessary undertaking beyond what this paper attempts: it requires risk-category ground truth, confidence-threshold behavior under realistic model uncertainty, and outcome data for each of the five decision-matrix actions in **Figure 3**, none of which the binary-detection experiment in Sections 5 and 6 was designed to produce.

9. Conclusion

This paper set out to argue for a specific reframing: a laboratory interoperability exception is not a bare technical failure but a risk signal waiting to be recognized as one and recognizing it that way takes four things working together—deterministic validation, machine-learning anomaly detection, semantic knowledge, and

human-governed resolution. Sections 4 through 7 developed each of those four capabilities in turn and showed how they compose: validation and anomaly detection catch what a rules engine or a classifier alone would miss between them; semantic knowledge turns a bare flag into an explanation; and human-governed resolution, guided by a risk score kept strictly independent of the AI's own confidence, decides what happens next—never the AI unilaterally. The synthetic evaluation in Section 6 gave the first part of that argument, hybrid detection over any single method, a first illustrative empirical test (F1 of 0.983 against 0.278 - 0.833), and Section 8 was plain that this is a demonstration on constructed data, not a clinical finding. What we believe holds regardless of that limitation is narrower and, we think, worth building toward: that a laboratory exception is never just a technical failure, that recognizing it as a risk signal requires all four capabilities rather than any one alone, and that keeping an AI system's confidence in itself separate from its assessment of danger is not a detail but the discipline that makes such a system safe to deploy at all.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Luu, H.S., Campbell, W.S., Cholan, R.A., Edgerton, M.E., Englund, A., Keller, A., *et al.* (2024) Analysis of Laboratory Data Transmission between Two Healthcare Institutions Using a Widely Used Point-to-Point Health Information Exchange Platform: A Case Report. *JAMIA Open*, **7**, ooae032. <https://doi.org/10.1093/jamiaopen/ooae032>
- [2] U.S. Food and Drug Administration (2024) Systemic Harmonization and Interoperability Enhancement for Laboratory Data (SHIELD). <https://www.fda.gov/medical-devices/diagnostic-data-program/systemic-harmonization-and-interoperability-enhancement-laboratory-data-shield>
- [3] Strawley, C., Larson, J. and Patel, V. (2024) Laboratory Interoperability through Health Information Exchange Organizations (ASTP/ONC Data Brief No. 74). Office of the Assistant Secretary for Technology Policy. <https://www.healthit.gov/data/data-briefs/laboratory-interoperability-through-health-information-exchange-organizations/>
- [4] HL7 International. FHIR (Fast Healthcare Interoperability Resources) Specification. <https://www.hl7.org/fhir/>
- [5] Regenstrief Institute. Logical Observation Identifiers Names and Codes (LOINC). <https://loinc.org/>
- [6] SNOMED International. SNOMED CT. https://docs.snomed.org/snomed-ct-specifications/snomed-ct-editorial-guide/readme/authoring/domain-specific-modeling/observable-entity/observable-entity-modeling/observable-entity-and-microbiology-test-results?utm_source=chatgpt.com
- [7] Chatterjee, A., Pahari, N. and Prinz, A. (2022) HL7 FHIR with SNOMED-CT to Achieve Semantic and Structural Interoperability in Personal Health Data: A Proof-of-Concept Study. *Sensors*, **22**, Article 3756. <https://doi.org/10.3390/s22103756>

- [8] International Organization for Standardization (2022) ISO 15189:2022: Medical Laboratories—Requirements for Quality and Competence. 4th Edition. ISO.
<https://www.iso.org/standard/15189.html>
- [9] Plebani, M. (2007) Laboratory Errors: How to Improve Pre- and Post-Analytical Phases? *Biochemia Medica*, **17**, 5-9. <https://doi.org/10.11613/bm.2007.001>
- [10] Najat, D. (2017) Prevalence of Pre-Analytical Errors in Clinical Chemistry Diagnostic Labs in Sulaimani City of Iraqi Kurdistan.
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5249186/>
- [11] Nassiri, O., Bousnina, F. and Hami, A. (2024) Non-Conformities in the Pre-Analytical Phase at the Parasitology-Mycology Laboratory of the Mohammed VI University Hospital in Oujda. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11257732/>
- [12] Çubukçu, H.C. (2026) The Role of AI in the Pre-Analytical Phase-Use Cases. *Clinical Chemistry and Laboratory Medicine*, **64**, 4-21.
<https://doi.org/10.1515/cclm-2025-1220>
- [13] Automation in Pre-Analytical Phase of Lab Testing and Use of AI in Lab Quality Control [Slide Presentation].
<https://www.slideshare.net/slideshow/automation-in-pre-analytical-phase-of-lab-testing-and-use-of-ai-in-lab-quality-control/286031380>
- [14] Chan, C.W. (2025) Data Analytics for Error Detection in Clinical Laboratories. *Critical Reviews in Clinical Laboratory Sciences*, **63**, 220-236.
<https://pubmed.ncbi.nlm.nih.gov/40970626/>
- [15] MLO Online (2026) Artificial Intelligence in Laboratory Information Systems: Transforming the Future of Clinical Diagnostics.
<https://www.mlo-online.com/information-technology/lis/article/55390593/artificial-intelligence-in-laboratory-information-systems-transforming-the-future-of-clinical-diagnostics>
- [16] Centers for Disease Control and Prevention (2026) National Notifiable Diseases Surveillance System (NNDSS). <https://www.cdc.gov/nndss/>
- [17] Shang, Y., Tian, Y., Lyu, K., Zhou, T., Zhang, P., Chen, J., *et al.* (2024) Electronic Health Record-Oriented Knowledge Graph System for Collaborative Clinical Decision Support Using Multicenter Fragmented Medical Data: Design and Application Study. *Journal of Medical Internet Research*, **26**, e54263.
<https://doi.org/10.2196/54263>
- [18] Jiang, P., Xiao, C., Jiang, M., Bhatia, P., Kass-Hout, T., Sun, J. and Han, J. (2024) Reasoning-Enhanced Healthcare Predictions with Knowledge Graph Community Retrieval. arXiv. <https://arxiv.org/abs/2410.04585>
- [19] Sherafatmandjoo, H., Akbari, M. and Rahmati, Z. (2026) Semantic Reasoning in Medicine: The Role of Knowledge Graphs across Five Key Domains. arXiv.
<https://arxiv.org/abs/2606.15155>
- [20] Wang, Z.D., Shi, D.Q., Zhao, J.Y., *et al.* (2025) Automated Construction of Medical Indicator Knowledge Graphs Using Retrieval Augmented Large Language Models.
<https://arxiv.org/pdf/2511.13526>