Event-Driven Clinical Decision Support: Securing Medical Logic through Structural Fault Isolation

Moritz GROB[a,b,1], Leonhard HAUPTFELD[b], Julia LIEPOLD[b,c], Julia ZECKL[b], Andrea RAPPELSBERGER[a], and Klaus-Peter ADLASSNIG[a,b]
[a]*Medical University of Vienna, Center for Medical Data Science, Institute of Artificial Intelligence, Spitalgasse 23, 1090 Vienna, Austria*
[b]*Medexter Healthcare, Borschkegasse 7/5, 1090 Vienna, Austria*
[c]*TU Wien, Institute for Logic and Computation, 1040 Vienna, Austria*
ORCiD ID: Moritz Grob 0009-0006-7867-9465, Leonhard Hauptfeld 0009-0006-5902-4666, Julia Liepold 0009-0004-5416-6417, Andrea Rappelsberger 0009-0001-0345-4011, Klaus-Peter Adlassnig 0009-0008-5046-1549

Abstract. Monolithic clinical decision support architectures entangle IT infrastructure with medical logic, compounding regulatory validation overhead (e.g., IEC 82304-1). We present an event-driven microservice architecture designed to streamline the clinical validation process through structural fault containment. Orchestrated by a Kafka message broker and gated by a Drools rule engine, the system intercepts structurally invalid data before it reaches the deterministic medical logic core comprised of ArdenSuite. Empirical testing demonstrated zero state loss during node failures and confirmed that shedding malformed data at the architectural perimeter actively prevents computational bottlenecks, effectively reducing the mean evaluation time per valid input during high-stress scenarios. By enforcing strict architectural boundaries, the pipeline decouples IT reliability from clinical safety. This confines the rigorous clinical validation burden entirely to the isolated clinical interpretation engine. The resulting secure containment boundary accelerates the integration of diverse data ingestion modalities and the continuous adaptation of clinical logic, equipping healthcare systems with an agile, compliant foundation for evolving decision support.

Keywords. Clinical Decision Support Systems, Software Architecture, Microservices, Event-Driven Architecture, Arden Syntax, Apache Kafka

1. Introduction

Modern clinical decision support (CDS) systems require the rapid and precise interpretation of multi-parametric clinical data [1]. Historically, monolithic CDS platforms tightly coupled data ingestion, syntactic pre-processing, and clinical logic execution, introducing critical failures in scalability and system assurance [1].

Under strict health regulations (e.g., IEC 82304-1), this entanglement creates an unsustainable validation burden, as minor technical modifications necessitate a complete re-validation of the entire clinical system [2].

[1] Corresponding Author: Moritz Grob; E-mail: mg@medexter.com

Event-driven microservice architectures, orchestrated by immutable message brokers, offer a paradigm shift [3]. Decoupling services into independent asynchronous nodes can facilitate high throughput as well as robust fault isolation [4, 5]. However, integrating these distributed IT patterns with the strict deterministic execution requirements of clinical knowledge engines remains a complex challenge.

This paper demonstrates a Kafka-backed, Drools-gated microservice pipeline shielding the ArdenSuite core of *iKnowledgeLab*. We establish how enforcing absolute architectural boundaries between data ingestion, structural validation gating, and knowledge-based interpretation (e.g., via ArdenSuite [6]) isolates the validation burden.

By intercepting IT-level faults and structurally invalid data at the periphery, the architecture strictly decouples the IT infrastructure from the medical logic. In practical terms, this isolation ensures that technical updates to the data ingestion pipelines or message brokers do not force a regulatory re-certification of the clinical diagnostic rules. This separation of validation burdens is a structural prerequisite for the agile and legally compliant integration of external data extraction tools.

Ultimately, this structural decoupling serves the clinical end-user. By guaranteeing that CDS interfaces remain uninterrupted by backend IT faults, the architecture preserves clinician trust, prevents alert fatigue caused by system-level errors, and ensures the continuous availability of critical diagnostic insights at the point of care.

2. Methods

2.1. Architectural Paradigm

iKnowledgeLab utilizes an event-driven microservice architecture detailed in Figure 1 to decouple data ingestion, validation, and medical logic execution. A central Job Manager aggregates system state, while a message broker organizes data into topics, each representing a distinct processing stage in the payload lifecycle. Microservices consume data exclusively from their designated topics, process the payload, and output to the subsequent state topic, guaranteeing structural fault isolation [5].

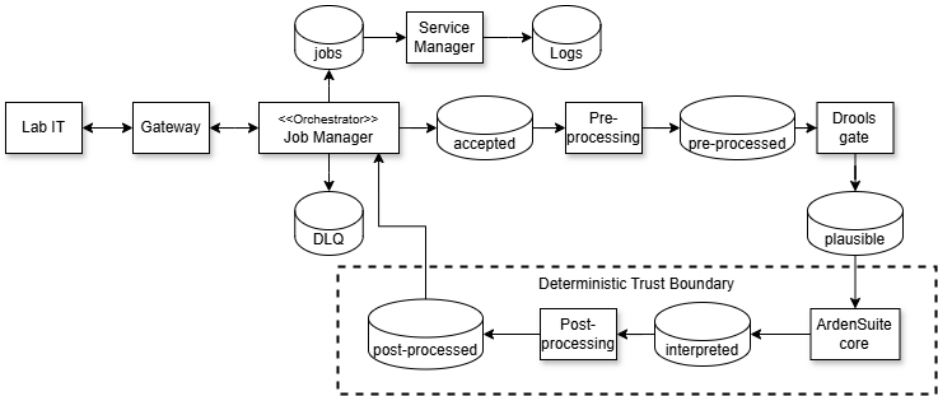

Figure 1. Flowchart of the event-driven microservice architecture, detailing the separation of state topics (circles) and processing services (rectangles). DLQ = Dead Letter Queue.

2.2. Pre-Processing and Syntactic Validation

Incoming data traverses a rigid sanitization pipeline. First, the Job Manager syntactically validates the JSON payload's structural integrity and data types. The pre-processing service then standardizes raw inputs (e.g., converting dates into computable age metrics) and extracts categorical features from unstructured remarks via regular expressions.

A deterministic rule engine (Drools) then executes strict boundary checks, enforcing unit compliance, required parameter presence, and hardcoded numerical limits. Gross syntactic violations detected by the Job Manager are routed directly to a Dead Letter Queue (DLQ). Conversely, payloads violating the Drools boundary checks are written to the subsequent state topic with an explicit error flag. This dual mechanism fully shields the downstream ArdenSuite core, which is configured to exclusively consume payloads with a valid status.

From a user-centric perspective, this rigid sanitization guarantees that the clinician is only ever presented with alerts derived from structurally sound and medically plausible data, effectively insulating the user from raw infrastructural errors.

2.3. The Deterministic Core

Only validated payloads reach the ArdenSuite platform. The interpretation of these complex arrays is executed via self-contained Medical Logic Modules (MLMs) written in HL7 Arden Syntax [7]. These MLMs encapsulate discrete clinical rules using explicit data, logic, and action slots, processing the structured parameters through strictly defined knowledge-based reasoning. This ensures that the medical logic execution remains fully deterministic, transparent, and strictly aligned with validated medical guidelines [6].

3. Results

3.1. Fault Tolerance and State Recovery

The architecture's reliance on a message broker successfully decouples service availability from data integrity, proven across two primary fault scenarios:

Processing Node Failure: If a processing node crashed mid-evaluation, the synchronous REST-request initiated by the external laboratory system times out, but internal state loss is entirely prevented. The payload remains securely persisted in the message broker's preceding topic (e.g., *pre-processed*). Upon node restart, the service immediately consumes the pending payload, resuming the pipeline without requiring data resubmission from the client.

Observability and Logging Failure: During a documented crash of the central Service Manager responsible for database logging, the pipeline's core clinical processing continued uninterrupted. While real-time telemetry was temporarily severed, the integrity of the audit trail was preserved. The message broker’s *jobs* topic acts as a persistent buffer, configured with log compaction and a seven-day retention policy. Upon restoration, all buffered state transitions were retroactively ingested into the metrics database, ensuring unbroken traceability.

3.2. Structural Fault Containment and Performance under Stress

The Drools gating mechanism successfully isolated ingestion errors without invoking the ArdenSuite execution core. To empirically validate the efficiency of this fault containment, the architecture was subjected to stress testing under varying degrees of data degradation on a local testing environment. The objective was to measure the mean evaluation time per valid input to determine if the processing overhead from isolating malformed data negatively impacted system performance.

The results, depicted in Table 1, indicate that the structural decoupling effectively insulates the deterministic core. Counterintuitively, the mean evaluation time per valid input decreased during high-stress scenarios. This occurs because the architectural perimeter (Job Manager and Drools) intercepts and discards malformed payloads almost instantly. By shedding invalid data early, the architecture prevents computational bottlenecks, ensuring the ArdenSuite core expends resources exclusively on viable medical logic without being penalized by infrastructural noise.

Table 1. System performance under varying fault injection loads (Simulated batches of 1,000 payloads).

Test Scenario	Fault Rate	Primary Interception Point	Mean Evaluation Time per Valid Input
Baseline	0%	n/a	10.75 s
High Syntactic Noise	50%	Job Manager (DLQ)	8.82 s
Boundary Violations	50%	Drools Gate	8.78 s
Mixed Extreme Stress	80%	Job Manager & Drools Gate	9.71 s

4. Discussion

While the architectural mechanisms described are inherently technical, their primary utility is deeply user-centric. In high-stakes clinical environments, users rely on CDS systems for accurate decision-making. When IT failures or malformed data propagate to the user interface, they erode clinical trust and disrupt workflows. By confining structural faults entirely to the IT layer, this architecture ensures that the clinician's interaction remains seamless, reliable, and strictly focused on vetted medical logic.

Transitioning from monolithic to event-driven CDS introduces regulatory challenges regarding determinism and state management. While empirical results confirm high availability and state recovery, these metrics validate system reliability rather than clinical safety.

Regarding health software validation (e.g., IEC 82304-1 [2]), the architecture ensures structural fault containment. Diverting malformed payloads to a DLQ and enforcing strict state-based filtering for boundary violations guarantees the deterministic ArdenSuite medical logic core executes exclusively on valid data that meets predefined criteria. This boundary acts as a technical gatekeeper; it does not inherently mitigate hazards from plausible but clinically inaccurate data. The value of this isolation lies in decoupling IT infrastructure validation from clinical logic validation, thereby narrowing the scope of the safety case for the clinical interpretation engine.

The design supports cybersecurity standards (e.g., IEC 81001-5-1 [8]) via defense-in-depth, eliminating point-to-point API vulnerabilities while preserving an immutable audit trail [5]. This modular decoupling is critical for future system evolution. For example, directly integrating Large Language Models (LLMs) to extract structured inputs for MLMs [9] into the interpretation engine would risk polluting the deterministic

safety boundary. The proposed architecture resolves this by enabling the hosting of probabilistic extraction pipelines as isolated producer nodes. Channeling stochastic outputs through the Drools gate strictly confines probabilistic risks, isolating the validation burden to independent mitigation layers, such as human-in-the-loop workflows.

Furthermore, this modular decoupling establishes the regulatory foundation for clinical agility within healthcare systems. Institutions can systematically integrate new or updated MLMs to expand the CDS scope, adapting rapidly to evolving medical guidelines or localized care pathways. Because the clinical execution core is isolated from ingestion pipelines, integrating expanded capabilities does not necessitate a systemic revalidation of the underlying IT infrastructure.

5. Conclusion

iKnowledgeLab demonstrates that an event-driven microservice architecture, gated by deterministic boundary rules, resolves the tension between distributed scalability and structural fault isolation. Enforcing absolute architectural boundaries between data ingestion, pre-processing, and medical logic execution provides a robust foundation for regulatory compliance. By creating modular, isolated risk zones, the architecture confines the rigorous clinical validation burden entirely to the interpretation engine. This ensures that future extensions—from external data pipelines to new clinical knowledge modules—can be deployed safely, equipping healthcare systems with continuously adaptable decision support.

References

[1] Sutton RT, Pincock D, Baumgart DC, Sadowski DC, Fedorak RN, Kroeker KI. An overview of clinical decision support systems: benefits, risks, and strategies for success. npj Digit Med. 2020;3:17. doi: 10.1038/s41746-020-0221-y

[2] Toivakka H, Granlund T, Poranen T, Zhang Z. Towards RegOps: A DevOps Pipeline for Medical Device Software. Lect Notes Comput Sci. 2021;13126:290–306. doi: 10.1007/978-3-030-91452-3_20

[3] Pallaprolu S. Event-driven architecture: A modern paradigm for real-time responsive systems. World J Advanced Engineering Technology and Sciences. 2025;15(2):2860–7. doi: https://doi.org/10.30574/wjaets.2025.15.2.0826

[4] Mhatre AL. Microservices Architecture for Healthcare. J Artif Intell Mach Learn Data Sci. 2023;1(4):1590–2. doi: 10.51219/jaimld/anand-laxman-mhatre/356

[5] Owobu WO, Abieba OA, Gbenle P, Onoja JP, Daraojimba AI, Adepoju AH, Chibunna UB. An End-to-End Pipeline Model for Real-Time Monitoring and Adaptive Fault Recovery in Kafka-Backed Microservice Environments. Int J Acad Manag Sci Res. 2025;9(4):290-318.

[6] Grob M, Hauptfeld L, Rappelsberger A, Adlassnig K-P. Standards-Based Interoperability for Clinical Decision Support Systems. Stud Health Technol Inform. 2025;327:173–7. doi: 10.3233/shti250296

[7] Health Level Seven International. Health Level Seven Arden Syntax for Medical Logic Systems, Version 3.0 STU2 [Internet]. Ann Arbor (MI), USA: HL7 International; 2025 [cited 2026 Mar 20]. Available from: https://www.hl7.org/implement/standards/product_brief.cfm?product_id=639

[8] Puder A, Henle J, Sax E. Threat Assessment and Risk Analysis (TARA) for Interoperable Medical Devices in the Operating Room Inspired by the Automotive Industry. Healthcare. 2023;11(6):872. doi: 10.3390/healthcare11060872

[9] Hauptfeld L, Grob M, Liepold L, Rappelsberger A, Adlassnig K-P. Knowledge-Based Interpretation of Multi-Modal Clinical Findings: Evaluating a Local Agentic Bridge Between Worlds. Proc Med Inform Eur (MIE) 2026. In: Stud Health Technol Inform. Forthcoming 2026 May.