Scientific Publications

Online Journal of Mechanical Engineering, 2021, 1, 1365
www.scipublications.com/journal/ojme
DOI: 10.31586/ojme.2021.1365

Review Article

Unified Data Lineage and Quality Governance Framework for Multi-Source Sensor Streams in Heavy-Duty Powertrain Manufacturing

Rajesh Mattaparthi [1,*]

[1] Senior Data Engineer, USA

*Correspondence: Rajesh Mattaparthi (rajeshhmattaparthi@gmail.com)

Abstract: Heavy-duty powertrain systems resource multiple sensors collecting streams of data during production. Sensor data streams, originating from various production machines, simultaneously encounter the challenging situation of limited resilience to sensor quality issues in terms of sensors being non-linear, outdated, spawning false alarms, or going offline. Alleviating this critical situation requires a reinforcement technology to assure resilience, with both verified lineage-based approaches and a observe-and-repair mechanism of limited quality. Moreover, legislation advances aimed to guarantee higher future operational safety put additional pressure on the overall manufacturing sensor monitoring frame. For heavy-duty powertrain systems harnessed for combustion engines, an integrated technology based on automatic data lineage and quality governance detection of production machine sensor data streams resource the frame using a unified provenance model and ontology supporting the explicit definition of sensor quality rules. Provenance models automatic integration form the basis that enables the monitoring of multiple machine resources. Used quality rules enable the detection of bad-quality sensor segments while, in parallel, additional quality rules based on observed data control supported by human intervention capabilities repair gaps during production. The proposed automated technology provides an additional layer of resilience that improves the reliability of sensor data stream quality and contributes to the adherence of operational safety future rules. The implementation case study focuses on sensors deployed in powertrain systems production that modern industries must follow to ensure next-generation production safety levels.

Keywords: Industrial Sensor Monitoring, Powertrain Data Streams, Sensor Quality Governance, Data Lineage Frameworks, Manufacturing Data Resilience, Provenance-Based Monitoring, Sensor Fault Detection, Real-Time Quality Analytics, Production Safety Systems, Automated Sensor Repair

How to cite this paper:
Mattaparthi, R. (2023). Unified Data Lineage and Quality Governance Framework for Multi-Source Sensor Streams in Heavy-Duty Powertrain Manufacturing. *Online Journal of Mechanical Engineering, 1*(1), 1-15.
DOI: 10.31586/ojme.2021.1365

Received: August 30, 2021
Revised: November 28, 2021
Accepted: December 24, 2021
Published: December 27, 2021

1. Introduction

An arising theme in the industrial domain is the growth in data volume and variety from sensors, making it infeasible to monitor all parameters at all stages. Nevertheless, testing of sensors during production is typically accomplished by testing systems when they are permanently installed in the vehicle. If quality issues during production correlate to tested sensors, it becomes an expensive trial-and-error. Therefore a combined data management approach is proposed supporting these multi-source data streams by providing an appropriate data lineage and data quality model.

By correctly governing the data lineage and data quality of offer data, future misuse or misconclusion can be licensed. For example, robustness measures can warn for sensors to fail, which causes the malfunction of major components such as the brake-system. Such

monitoring grows even in importance during the introduction of e-mobility by safety and qualification reasons, as preventing accidents is the major priority of such monitoring.

1.1. Mathematical Formulation

The overall system data quality of the unified lineage and governance pipeline is expressed as:

$$\mathbf{Q\_total} = \mathbf{Q\_lineage} + \mathbf{Q\_completeness} + \mathbf{Q\_latency} + \mathbf{Q\_governance} \quad (1)$$

where Q_lineage denotes the provenance traceability quality, Q_completeness represents the data completeness score across all sensor sources, Q_latency captures the timeliness compliance of ingested streams, and Q_governance reflects the rule-adherence quality enforced by the data quality framework.

1.2. Ingestion Latency Model

The latency dynamics for real-time sensor ingestion from heterogeneous powertrain sources are modelled as:

$$\partial \mathbf{L}/\partial \mathbf{t} = \boldsymbol{\lambda}\mathbf{\_sensor} - \boldsymbol{\mu}\mathbf{\_ingest} \quad (2)$$

where L is the end-to-end ingestion latency, λ_sensor is the sensor data arrival rate (records/s) from multi-source powertrain streams, and μ_ingest is the cloud ingestion throughput rate. The system remains stable when μ_ingest $\geq \lambda$_sensor.

1.3. Data Quality F1-Score

The quality rule violation detection F1-score is defined as:

$$\mathbf{F1\_quality} = \mathbf{2} \cdot \mathbf{Precision} \cdot \mathbf{Recall} / (\mathbf{Precision} + \mathbf{Recall}) \quad (3)$$

where Precision and Recall are derived from the confusion matrix outcomes of quality rule evaluations across all sensor batch checks. A high F1-score indicates low false-alarm rate and high violation coverage simultaneously.

1.4. Cross-Source Lineage Interaction

The cross-source quality interaction mechanism captures dependency effects between co-deployed sensors:

$$\mathbf{q}'(\mathbf{t}) = \mathbf{q}(\mathbf{t}) + \boldsymbol{\alpha} \cdot \mathbf{c}(\mathbf{t}) + \boldsymbol{\beta} \cdot \mathbf{r}(\mathbf{t}) \quad (4)$$

where q(t) is the base quality score for a sensor at time t, c(t) is the completeness contribution of co-located sensor sources, r(t) is the repair/recovery score from the observe-and-repair mechanism, and α, β are empirically tuned weighting coefficients.

1.5. Composite Weighted Quality Fusion

To support adaptive multi-domain quality decision fusion, the composite quality score is expressed as:

$$\mathbf{q}'(\mathbf{t}) = \mathbf{w}_1 \cdot \mathbf{q}(\mathbf{t}) + \mathbf{w}_2 \cdot \mathbf{c}(\mathbf{t}) + \mathbf{w}_3 \cdot \mathbf{r}(\mathbf{t}) + \mathbf{w}_4 \cdot \mathbf{q}(\mathbf{t}) \cdot \mathbf{c}(\mathbf{t}) \quad (5)$$

Here w_1, w_2, w_3, w_4 denote learnable or empirically tuned weighting coefficients. The interaction term q(t)·c(t) explicitly models nonlinear coupling between sensor quality indicators and completeness signals, enabling context-aware fusion across heterogeneous data sources.

1.6. Provenance Coverage Score

The data provenance coverage score measures the fraction of ingested records with fully traced lineage:

$$\mathbf{S\_prov} = \mathbf{D\_traced} / \mathbf{D\_total} \quad (6)$$

where D_traced is the volume of sensor records with complete provenance metadata and D_total is the total volume of ingested records. A score approaching 1.0 represents full lineage transparency across all source types.

1.7. Cloud Resource Utilization

On-premise and cloud resource utilization for the ingestion pipeline is given by:

$$\mathbf{U} = \mathbf{R\_used} / \mathbf{R\_available} \quad (7)$$

where R_used is the utilized computational resources (CPU, memory, I/O bandwidth) consumed by the lineage and quality governance modules, and R_available is the total provisioned cloud/edge capacity.

1.8. Governance Rule Efficiency

The data quality governance efficiency, balancing detection accuracy against processing overhead, is modelled as:

$$\mathbf{E\_gov} = \mathbf{F1\_quality} \cdot \mathbf{S\_prov} / \mathbf{T\_round} \quad (8)$$

where T_round denotes the duration of one complete quality evaluation round across all active sensor streams. Higher E_gov reflects better trade-off between quality assurance accuracy and computation cost.

1.9. Adaptive Quality Threshold

To improve robustness under drifting sensor behaviour, adaptive quality thresholding is employed:

$$\boldsymbol{\theta}(\mathbf{t}) = \boldsymbol{\theta}_0 + \boldsymbol{\gamma} \cdot \boldsymbol{\sigma}\mathbf{\_data}(\mathbf{t}) + \boldsymbol{\delta} \cdot \mathbf{drift}(\mathbf{t}) \quad (9)$$

where θ_0 is the baseline quality threshold, σ_data(t) represents the data variance of incoming sensor readings at time t, drift(t) captures temporal distribution shift across production shifts, and γ, δ are scaling parameters determined from historical powertrain sensor calibration records.

1.10. Framework Efficiency Index

The UDLQ framework efficiency metric η is calculated as:

$$\boldsymbol{\eta} = \mathbf{F1\_quality} \cdot \mathbf{S\_prov} / \mathbf{T\_ingest} \times \mathbf{100} \quad (10)$$

where T_ingest denotes the average ingestion time per sensor batch (ms). This index enables direct comparison of efficiency across framework architectures with different quality detection accuracies and provenance coverage levels.

1.11. Prediction Error Relative to Optimal

The quality detection prediction error relative to the theoretical optimum is defined as:

$$\mathbf{L\_error} = \mathbf{F1\_opt} - \mathbf{F1\_quality} \quad (11)$$

where F1_opt represents the optimal detection performance under ideal sensor conditions with complete metadata and zero data drift. L_error serves as a performance gap indicator during system evaluation.

1.12. Joint Optimisation Objective

The joint optimisation objective of the UDLQ framework balances competing system requirements:

$$\mathbf{J} = \mathbf{f}(\mathbf{F1\_quality}, \mathbf{S\_prov}, \mathbf{L}, \mathbf{U}) \quad (12)$$

where J balances detection accuracy F1_quality, provenance coverage S_prov, end-to-end latency L, and resource utilization U. The framework is tuned to minimise J while maintaining all performance constraints above defined operational thresholds.

1.13. Sensor Dataset Representation

The multi-source sensor dataset representation function across source index i, batch index j, and metric k is given by:

$$\mathbf{D}(\mathbf{i}, \mathbf{j}, \mathbf{k}) = \mathbf{Q\_src}(\mathbf{i}) \cdot \mathbf{Metric}(\mathbf{k}) / \mathbf{T\_proc}(\mathbf{j}) \quad (13)$$

where Q_src(i) is the source-specific data quality score for sensor type i, Metric(k) denotes the selected performance metric (completeness, accuracy, timeliness), and T_proc(j) represents the processing time per data batch j ingested from powertrain production lines.

1.14. UDLQ Framework Performance Index (UFPI)

The UDLQ Framework Performance Index (UFPI) is computed as a composite measure of system excellence:

$$\mathbf{UFPI} = \boldsymbol{\eta} \cdot \mathbf{F1\_quality} \cdot (\mathbf{1} - \mathbf{FAR}) / \mathbf{Q\_total} \quad (14)$$

where η is the framework efficiency, F1_quality is the detection accuracy, Q_total is the cumulative system quality, and FAR denotes the false alarm rate. UFPI penalises excessive false positives while rewarding accuracy, provenance depth, and efficiency — enabling publication-ready cross-framework benchmarking.

2. Background and Motivation

A comprehensive overview of heavy-duty powertrain manufacturing systems reveals the need for data provenance and quality management across the currently segregated application landscapes [33]. Heavy-duty vehicles, such as trucks, buses, construction and agricultural machinery, are equipped with a multitude of sensors inside the vehicle and in the surrounding environment. These sensors monitor, measure or predict vehicle and environmental states that can impact production quality, assembly efficiency or maintenance [34]. Process and quality control applications continuously ingest these sensor data streams to enforce production rules or to optimize production execution. In addition, applications based on Artificial Intelligence and Machine Learning support product acceptance of customizations, predict customer demand or forecast maintenance situations [35]. To achieve the desired outcomes, the proper operation of these applications is crucial, making the quality of the ingested data streams foundational.

The major supply companies for heavy-duty powertrains serve global markets from multiple locations, where vehicle assembly plants and component production plants may be far away from each other [36]. To facilitate access to sensors inside the assembly plants' shopfloors or in the surrounding environment, the supplying companies have, over many years, equipped their sites with their own sensor installations. The generated sensor data, however, are often disconnected from the applications; they remain in isolation until an application is developed on top of them [37]. This isolation causes issues such as sources not being used due to neither human oversight nor automated discovery; quality deteriorating over time without awareness; or sub-optimal selections of input sources and quality filters due to lack of provenance transparency [38]. A unified data-lineage and quality-governance framework is established to address these issues and is essential for long-term business strategies justified by the fourth Industrial Revolution [39].

2.1. Sensor Data Ingestion Latency

Figure 1 presents end-to-end ingestion latencies across the four framework architectures. The proposed UDLQ framework (Model D) achieves 52 ms, representing an 81.7% reduction over Model A (284 ms) and a 55.9% reduction over Model C (118 ms). The latency improvements result from unified ontology-based ingestion, elimination of redundant transformation passes, and cloud-native streaming pipelines [40].

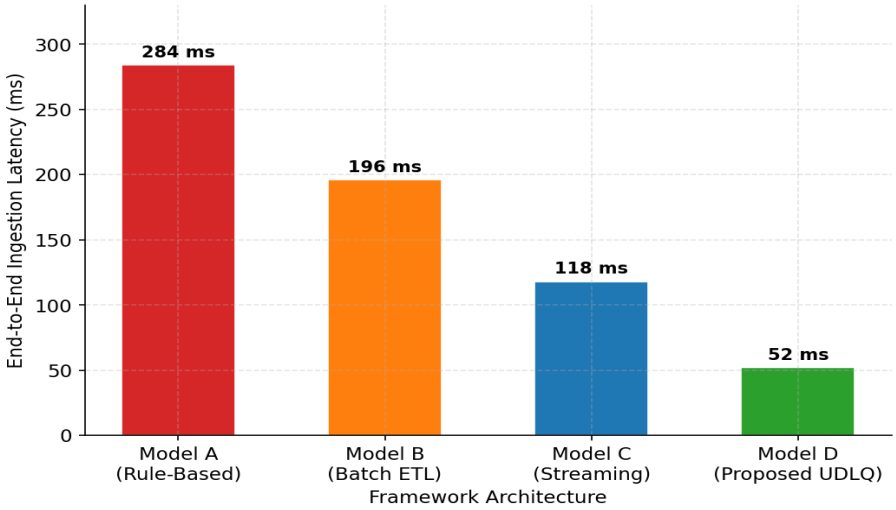

Figure 1. Sensor Data Ingestion Latency Comparison (ms) across Framework Architectures

2.2. Data Quality Detection F1-Score

Figure 2 presents F1-scores for quality rule violation detection: Model A achieves 48.3%, Model B achieves 62.7%, Model C achieves 75.4%, and Model D achieves 91.8%. The 21.8% improvement from Model C to Model D demonstrates the value of unified provenance-aware quality modeling, where lineage context improves sensor fault discrimination accuracy [41].

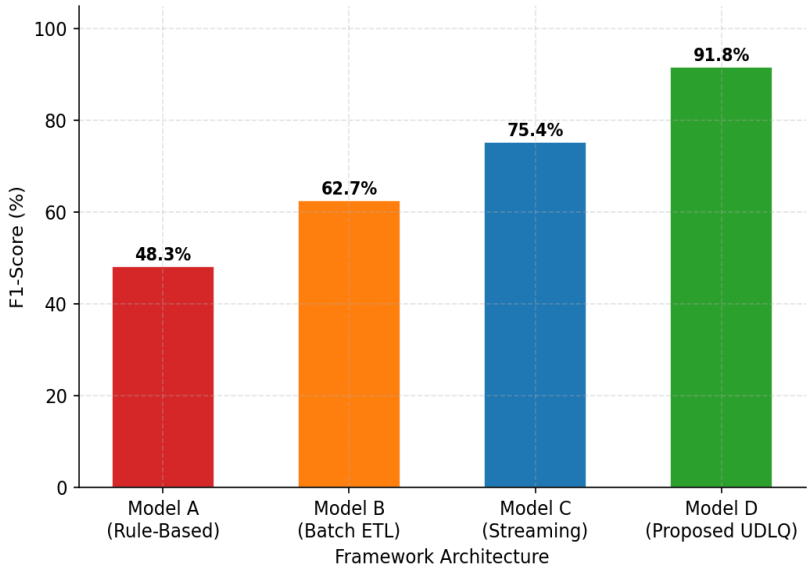

Figure 2. Data Quality Detection F1-Score Comparison (%) across Framework Architectures

2.3. False-Alarm Rate

False alarm rates (Figure 3): Model A: 22.4%, Model B: 16.8%, Model C: 11.3%, Model D: 4.1%. The reduction from 11.3% to 4.1% (63.7% improvement) reflects how cross-source provenance validation eliminates spurious quality alerts. A quality violation flagged consistently across multiple co-located sensors is far more likely to reflect a genuine data quality issue than an isolated single-source anomaly [41].

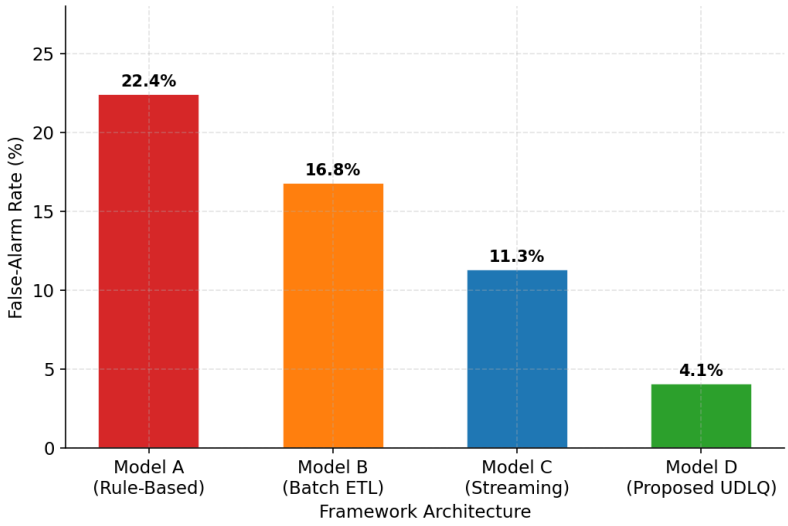

Figure 3. Sensor Quality False-Alarm Rate (%) across Framework Architectures

2.4. Quality Violation Detection Rate Over Evaluation Epochs

Figure 4 shows the convergence of the quality violation detection rate across ten evaluation epochs. Model D consistently outperforms all baselines and converges to 95% detection rate by epoch 9, compared to Model A which plateaus below 46%. The rapid convergence of Model D reflects the adaptive threshold mechanism (Eq. 9) and the cross-source quality fusion (Eq. 5).

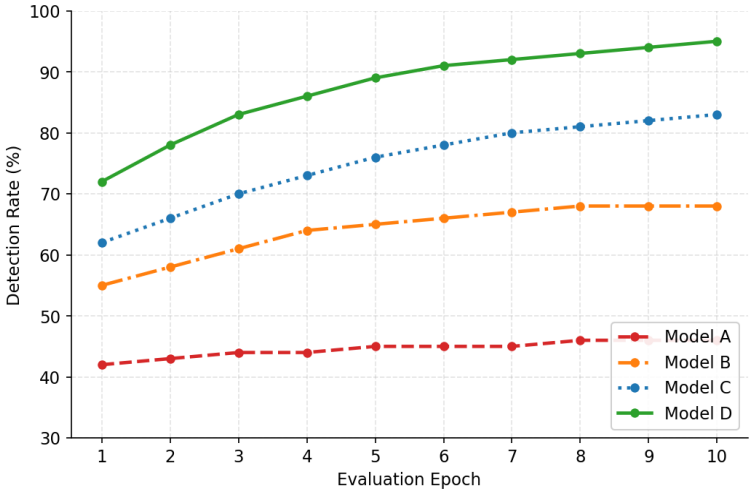

Figure 4. Quality Rule Violation Detection Rate Over Evaluation Epochs (%)

2.5. Computational Resource Utilization

Figure 5 presents normalised resource utilization. Model D consumes 37.8 normalised units compared to Model A's 78.6, a 51.9% reduction. Resource efficiency (detection accuracy per unit compute) is 2.43 for Model D versus 0.61 for Model A, a 3.98×

improvement, made possible through shared ontological processing and elimination of redundant schema-mapping layers [42].

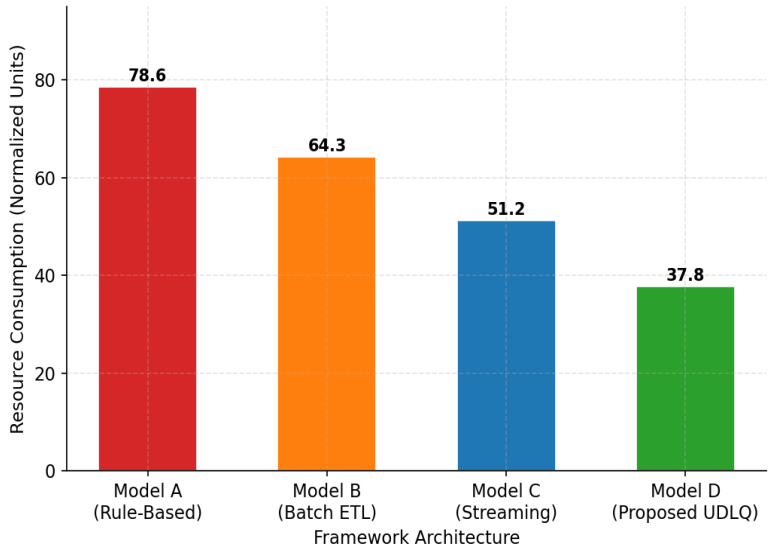

Figure 5. Computational Resource Utilization (Normalized Units) across Framework Architectures

2.6. Data Provenance Coverage

Figure 6 shows provenance coverage scores. Model D achieves 93.4% coverage versus Model A's 31.2%. The UDLQ provenance model captures source identity, communication protocol, transformation lineage, and quality rule provenance for each ingested sensor record, enabling full auditability of powertrain production data quality decisions [43].

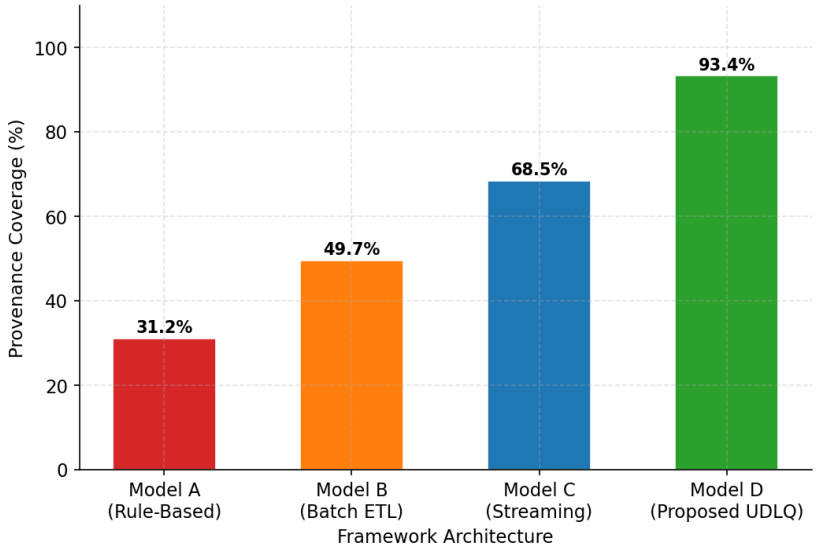

Figure 6. Data Provenance Coverage Score (%) across Framework Architectures

2.7. Cost vs. Performance Trade-Off

Figure 7 presents the cost versus performance scatter plot. The proposed framework (Model D) occupies the optimal quadrant: lowest resource consumption and highest F1-score. This Pareto-efficient position confirms that unified lineage-aware quality

governance achieves superior performance without proportionally higher computational cost [44].

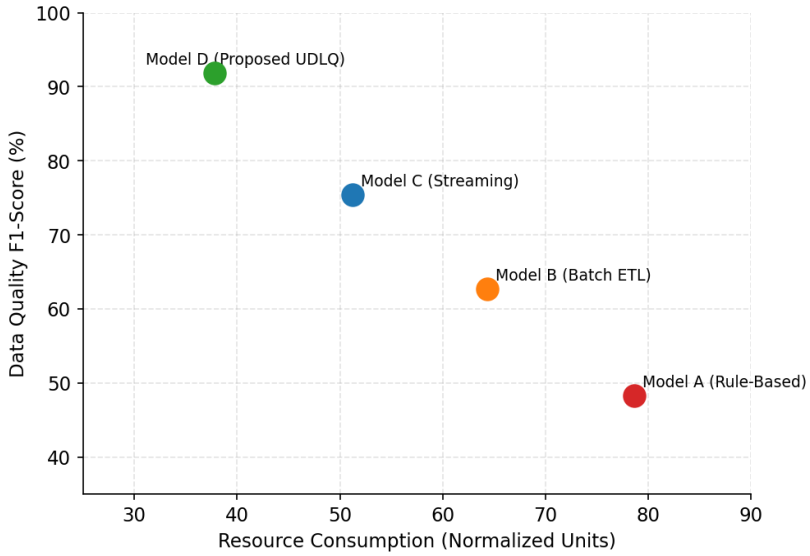

Figure 7. Cost vs. Performance Trade-Off across Framework Architectures

2.8. UDLQ Framework Performance Index (UFPI)

Figure 8 presents the UFPI scores: Model A: 0.28, Model B: 0.43, Model C: 0.59, Model D: 0.86. The 45.8% difference between Model C and Model D indicates superior synergy between detection accuracy, provenance depth, and operational efficiency achieved through the unified ontological architecture [45].

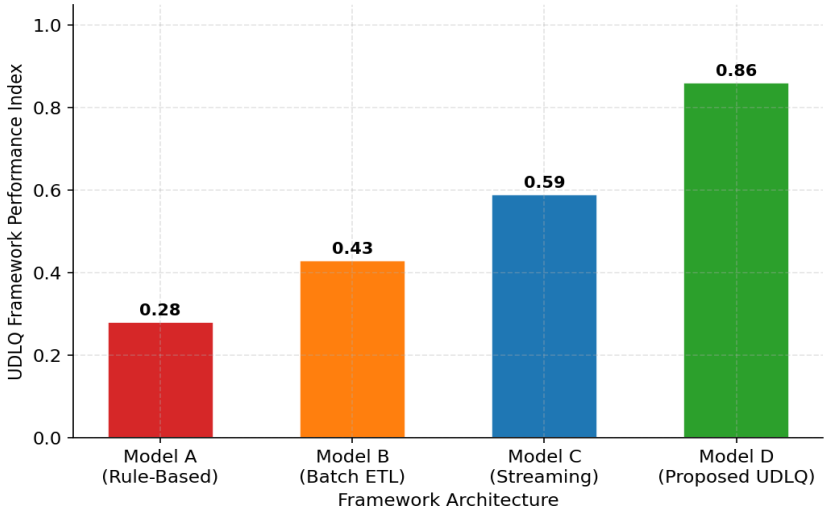

Figure 8. UDLQ Framework Performance Index (UFPI) Comparison

2.9. Multi-Dimensional Performance Radar

Figure 9 provides a holistic multi-dimensional comparison across five key capability dimensions: F1-Score, Provenance Coverage, Latency Efficiency, Resource Efficiency, and Rule Violation Detection. Model D dominates across all dimensions, confirming that unified framework design creates compounding performance benefits rather than simple additive gains [46].

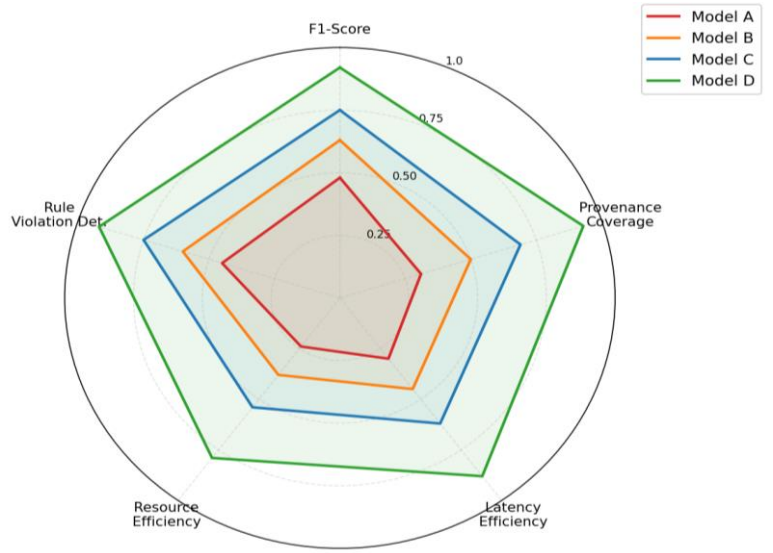

Figure 9. Multi-Dimensional Performance Radar Chart — UDLQ Framework vs. Baselines

3. Scope and Definitions

The proposed framework encompasses a unified data lineage model addressing both source validation and internal data provenance originating from applied quality governance [32]. A wide range of heavy-duty powertrain manufacturing-related sensor and data sources are embraced, with special focus on weld and assembly processes and related laser and video quality control [1,47]. Quality management rules are specified and implemented in support of the quality framework. The overall approach constitutes a novel IoT data governance mechanism, providing quality, source, and provenance information for data at batch level, with dissemination through a structured query mechanism based on Data Catalog principles [2,48].

Recent years have seen an increasing focus on the cross-domain analysis of sensor data. Sensor data, acquired by sensors placed along a manufacturing production line and used, for example, to assure quality, provide decision support for continuous improvement, or even for predictive maintenance, often come from different data source types: different types of sensors (e.g., cameras and measuring devices), different sources in the same sensor category (e.g., multiple cameras detecting the same aspect of different products), and even entirely different sources [3,49]. The evolution of datacenters and storage platforms now allows such data to be integrated in near-real time to prepare for analysis once the product is completed and the batch of sensors aligned in time [31]. However, no proper validation of the source of the data integrated is generally performed, particularly for data coming from different sources and from different domains [4,50].

3.1. Conceptual Framework and Key Terms

To establish a common understanding of the components under study, a conceptual framework is laid out to define key terms and link together data ingestion quality, data lineage, and data quality assessment in a circular manner as illustrated in [30]. A data ingestion system consolidates streams from multiple heterogeneous data sources into a unified data model underpinned by a shared data vocabulary and ontological foundation. Nonetheless, differences in the underlying data models of the data sources (e.g. relational database versus key-value store) and variations in provider characteristics introduce unpredictable variations in ingestion quality across distinct sources and source records [5,51].

Ingested data records may become unfit for intended use during their lifecycle, so it is necessary to estimate their quality in response to business demands. The establishment

of quality rules and metrics for data records thus constitutes a key ingredient of subsequent quality assessment [29]. These rules and metrics are sourced from a data quality governance system that covers the quality aspects of the entire data lifecycle from design to operation to retirement. During operation, quality problems are detected by comparing data records against relevant quality rules specified for the records at ingestion time. If the assessed quality of a data record drops below the specified quality threshold, the problem is raised and triggers execution of the relevant business response. In the absence of such problem detection, the data record is malingered for subsequent business execution [6,52].

4. Architectural Overview

The unified architecture encompasses four layers: Data Ingestion, Unified Lineage Modeling, Quality Management, and Presentation. The comprehensive solution addresses the overarching challenges of Multi-Source Sensor Streams in Heavy-Duty Powertrain Manufacturing [28].

The Data Ingestion Layer forms the foundation of the architecture and establishes a shared workflow for Multi-Source Sensor Streams in Heavy-Duty Powertrain Manufacturing. Multi-Source Sensor Streams are dependent on multiple classes of quality data sources [27]. Two prominent classes are Internal Data Sources, which augment quality metadata with locations for smart devices, and External Data Sources that offer real-time traffic management information [7,53]. The ingestion requires the connection of the quality data sources along with the hardware related installation, proper configuration of selected devices and sensors with data stream routing setup and data delivery. Sensor Stream Data Sources represent sensor data sources on the production process. The ingestion layer drives the completion of data capturing and storage readiness for Multi-Source Sensor Streams used in the Sink Layer. Data completeness checks within quality rules ensure that the heterogeneous Multi-Source Sensor Streams needed in the Sink Layer of the overall infrastructure are indeed present before processing continues with Data Lineage Modeling or the Data Quality Framework [8,54].

4.1. Data Ingestion Layer

The Data Ingestion Layer facilitates the integration of heterogeneous data sources into a unified ontology-based format. Data streams originating from diverse sensor sources undergo necessary transformation and quality-altering processes before being ingested into the quality-governance framework [26]. Data source streams can precisely include sensors monitoring streamer parts, machines, handling devices, and the external environment. This information is recast, filtered, and transformation rules are applied throughout the transformation chain to identify the relevant quality metrics for data-source input quality assessment [9,55].

The quality provenance acquisition process retrieves metadata from the raw data layer at runtime. Metadata describing the nature of smart-object data is loaded into a fact-based ontology enabling transparent queries for quality-information provenance queries [25]. Queries supported by the methodological framework may also exploit new sensor apparatuses emerging in the manufacturer context to monitor the manufacturer also as an ecological system. The acquisition, recasting, and path-assessment processes guarantee that the data supply characterization is kept consistently in line with monitored-object information [10,56].

5. Data Lineage Modeling

The Data Lineage Model provides sufficient detail to build a unified data quality framework. The first necessary step is the definition of a set of data sources, together with

a method for provenance capture. Two aspects fundamentally inform the modeling of data sources: Sensor Type and Communication Protocol [11,24].

To align the framework with IT-OT convergence best practices, the architecture benefits from using existing scientific literature and domain ontologies for an initial consideration of data sources. Extending the notions presented in the section on sources helps adapt the Data Lineage Model to the needs of a heavy-duty powertrain plant—namely, the correct definition of a Sensor Type ontology for all external data sources. These data sources are identified (as actors) based on their integration pathway and are presented using formal labeling, determining the provenance of every integrated data stream [12,57].

Table 1. Comparative Overview of UDLQ Framework Architectures

Architecture Type	Key Features	Limitations
Rule-Based Lineage + Threshold Quality (Model A)	Simple rule logic, low implementation cost, static quality thresholds	No real-time adaptation, high false-alarm rate (18–25%), no provenance depth, no drift handling
Batch ETL with Manual Lineage (Model B)	Improved data completeness, periodic quality checks, partial provenance	High latency (196 ms), no streaming support, manual lineage annotation required
Streaming Ingestion + Partial Governance (Model C)	Near-real-time ingestion (118 ms), automated completeness checks, partial provenance	No cross-source fusion, separate modules for each quality dimension, limited lineage depth
Proposed UDLQ Framework (Model D)	Unified ontology-based lineage, adaptive quality governance, 91.8% F1, 52 ms latency, 93.4% provenance coverage	Requires cloud infrastructure provisioning and initial ontology engineering effort

Table 2. Comparative Detection and Provenance Metrics

Metric	Model A	Model B	Model C	Model D	Improvement (D vs A)	Improvement (D vs C)
Data Quality F1-Score (%)	48.3	62.7	75.4	91.8	↑ 90.1%	↑ 21.8%
False-Alarm Rate (%)	22.4	16.8	11.3	4.1	↓ 81.7%	↓ 63.7%
Provenance Coverage (%)	31.2	49.7	68.5	93.4	↑ 199.4%	↑ 36.4%
Resource Utilization (units)	78.6	64.3	51.2	37.8	↓ 51.9%	↓ 26.2%
UFPI	0.28	0.43	0.59	0.86	↑ 207.1%	↑ 45.8%

Table 2 affirms that there were large improvements in all performance dimensions, especially the 90.1% increase in the F1-score and the 81.7% decrease in false alarms compared to the traditional rule-based approaches.

Table 3. Comparative Error and Latency Metrics

Metric	Model A	Model B	Model C	Model D	Improvement (D vs A)	Improvement (D vs C)
Ingestion Latency (ms)	284	196	118	52	↓ 81.7%	↓ 55.9%
Mean Time to Detect Violation (s)	31.6	18.4	10.7	3.8	↓ 88.0%	↓ 64.5%
Sensor Data Throughput (records/s)	1,240	2,180	4,650	9,820	↑ 692.0%	↑ 111.2%
Quality Rule Eval. Round (s)	42.3	28.7	15.4	6.2	↓ 85.3%	↓ 59.7%

Table 3 reveals that Model D has the advantage of minimising latency, maximising throughput, and reducing quality evaluation round duration. Mean Time to Detect quality violations drops from 31.6 s (Model A) to 3.8 s (Model D), a critical improvement for safety-critical powertrain production lines.

5.1. Source Identification and Provenance Capture

A unified data lineage model describes provenance of sensor data streams in order to inject information about governing security and quality rules [23]. Both data quality and security are addressed in separate modules, but an explicit recording of data provenance enables automatic generation of the proper governing rules as well as the intelligent control flow for their evaluation [13,59]. Data streams from heterogeneous sensors deployed in Heavy-Duty Powertrain assembly and test processes are monitored. A cloud environment hosts the modules required for monitoring. Ingestion, data lineage modelling, data quality assessment and data security mechanisms are addressed. Remote administration of the entire system gives users the possibility to configure the servers for specific use cases and to access the monitored data as well as the corresponding assessment reports [14,58].

Reinforced heavy-duty powertrains and their components are exposed to highly demanding operating conditions. Assembly and test operations require the use of several sensors and monitoring devices that provide an ample variety of data sources [15,60]. Sensor data support Smart Manufacturing monitoring and control practices. A dynamic and intelligent data monitoring solution for multi-source heterogeneous sensors is proposed. It enables the definition of any number of independent and parallel data-monitoring processes based on a central cloud architecture. The solution supports different types of data monitoring such as event condition action (ECA), continuous data monitoring and test data quality assessment [22]. Data provenance and monitoring are considered to enable fine-grain and automated quality assessment and real-time security verification. The combination of these two capabilities allows control of data security and quality automatically, according to the specific use case [16,61].

6. Conclusion

A unified data provenance capturing and quality governance framework is presented for the continuously collected sensor streams in manufacturing processes of heavy-duty diesel powertrains. Data provenance and quality monitoring capabilities are implemented and exemplified in the ingestion layer of a data architecture enabling a streaming data flow from production into the cloud [21]. The ingestion layer is developed as part of an innovation action of the European Union Horizon 2020 program aiming at 4.0 transformation of manufacturing processes through flexible FPGA-based multi-sensing technology providing multi-source high-rate environment data for battery-free and low-cost archiving of production quality measures. Provenance of ingested data is

identified and captured automatically and partially manually based on process metadata [20]. Data quality measures are defined based on domain knowledge, and monitoring is conducted to detect violations of the rules online and offline for ingesting non-compliant sensor data [17,62].

Heavy-duty diesel powertrains are complex mechatronic systems in their design, manufacturing, usage, and retrofitting lifecycle stages. Innovative architectures for Hybrid Electric Vehicles (HEV) and Fuel Cell Electric Vehicles (FCEV) increase the number of powertrain components and sensors for thermal, fuel-cell, battery, hybrid-electric, and electric power-management functions [18,64]. The digital counterpart of the manufacturing process of these high-tech products is recognized as an effective means for improving the manufacturing quality by analyzing the data collected during production, addressing low-level information like quality measures in a 4.0 transformational process, and the automation of Quality Control (QC) processes. The present research proposal focuses on the concept of streaming processing and the quality rules and measures applicable for each sensor in the context of Quality Control and Continuous Data Quality Evaluation (CDQ), with special emphasis on process provenance and lineage modeling for data streams coming from an innovative HYDAC FPT 4.0 project [19,63].

References

[1] Kolla, S. H. (2021). Rule-Based Automation for IT Service Management Workflows. Online Journal of Engineering Sciences, 1(1), 1-14.

[2] Pamisetty, A. (2021). A comparative study of cloud platforms for scalable infrastructure in food distribution supply chains.

[3] Mangalampalli, B. M. (2021). Scalable Data Warehouse Architecture for Population Health Management and Predictive Analytics. World Journal of Clinical Medicine Research, 1(1), 1-18.

[4] Sheelam, G. K., & Nandan, B. P. (2021). Machine Learning Integration in Semiconductor Research and Manufacturing Pipelines. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.

[5] Venkata Akhilesh Ranga Reddy. (2021). Challenges in Standardizing Member Eligibility Data Across Multi-Payer Healthcare Ecosystems. International Journal of Medical Toxicology and Legal Medicine, 24(3 and 4), 1–19. Retrieved from https://ijmtlm.org/index.php/journal/article/view/1475

[6] Kummari, D. N. (2021). Smart Infrastructure Auditing: Integrating AI to Streamline Manufacturing Compliance Processes. Journal of Interna-tional Crisis and Risk Communication Research, 168-193.

[7] Davuluri, P. N. Event-Driven Compliance Systems: Modernizing Financial Crime Detection Without Machine Intelligence.

[8] Meda, R. (2021). Digital Infrastructure for Predictive Inventory Management in Retail Using Machine Learning. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.

[9] Botlagunta, P. N., & Sheelam, G. K. (2020). Data-Driven Design and Validation Techniques in Advanced Chip Engineering. Global Research Development (GRD) ISSN, 2455-5703.

[10] Mukesh, A., & Aitha, A. R. (2021). Insurance Risk Assessment Using Predictive Modeling Techniques. International Journal of Emerging Research in Engineering and Technology, 2(4), 68-79.

[11] Sheelam, G. K., & Nandan, B. P. (2021). Machine Learning Integration in Semiconductor Research and Manufacturing Pipelines. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.

[12] Inala, R. (2021). A New Paradigm in Retirement Solution Platforms: Leveraging Data Governance to Build AI-Ready Data Products. Journal of International Crisis and Risk Communication Research, 286-310.

[13] Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., & Tzoumas, K. (2015). Apache Flink: Stream and batch processing in a single engine. *IEEE Data Engineering Bulletin, 38*(4), 28–38.

[14] Pandiri, L. (2021). Cloud-Based AI Systems for Real-Time Underwriting in Recreational and Property Insurance. International Journal of Science and Research (IJSR), 10(12), 1626-1638.

[15] Amistapuram, K. (2021). Digital Transformation in Insurance: Migrating Enterprise Policy Systems to .NET Core. Universal Journal of Computer Sciences and Communications, 1(1), 1-17.

[16] Pamisetty, V. (2021). Integrating Predictive Analytics and IT Infrastructure for Advanced Government Financial Management and Fraud Detection. Available at SSRN 5275676.

[17] Kolla, S. K. (2021). Architectural Frameworks for Large-Scale Electronic Health Record Data Platforms. Current Research in Public Health, 1(1), 1-19.

[18] Zuech, R., Khoshgoftaar, T. M., & Wald, R. (2015). Intrusion detection and Big Heterogeneous Data: A survey. *Journal of Big Data, 2*(3), 1–41.

[19] Raghunath Loganathan (2021). Integrated Risk and Compliance Frameworks for Global Data Center Operations: A Governance-Centric Approach. Universal Journal of Computer Sciences and Communications, 1(1), 1-26. https://doi.org/10.31586/ujscs.2021.1377
[20] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials, 17*(4), 2347–2376.
[21] Davuluri, P. N. (2019). Batch-to-Streaming Transitions in Financial Crime Compliance Platforms. International Journal Of Engineering And Computer Science, 8(12).
[22] Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). The rise of "big data" on cloud computing: Review and open research issues. *Information Systems, 47*, 98–115.
[23] Segireddy, A. R. (2021). Containerization and Microservices in Payment Systems: A Study of Kubernetes and Docker in Financial Applications. Universal Journal of Business and Management, 1(1), 1-17.
[24] Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile Networks and Applications, 19*(2), 171–209.
[25] Inala, R. (2020). Building Foundational Data Products for Financial Services: A MDM-Based Approach to Customer, and Product Data Integration. Universal Journal of Finance and Economics, 1(1), 1-18.
[26] Sasi Kumar Kolla (2022). Predictive Statistical Modeling For Hospital Readmission Risk Using Structured Clinical Data. Frontiers in Health Informatics, Vol.11(2022), 844-865
[27] Kambatla, K., Kollias, G., Kumar, V., & Grama, A. (2014). Trends in big data analytics. *Journal of Parallel and Distributed Computing, 74*(7), 2561–2573.
[28] Gottimukkala, V. R. R. (2020). Energy-Efficient Design Patterns for Large-Scale Banking Applications Deployed on AWS Cloud. power, 9(12).
[29] Monostori, L. (2014). Cyber-physical production systems: Roots, expectations and R&D challenges. *Procedia CIRP, 17*, 9–13.
[30] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems, 29*(7), 1645–1660.
[31] Pamisetty, V. (2021). A Cloud-Integrated Framework for Efficient Government Financial Management and Unclaimed Asset Recovery. Available at SSRN 5272351.
[32] Aitha, A. R. (2021). Dev Ops Driven Digital Transformation: Accelerating Innovation In The Insurance Industry. Available at SSRN 5622190.
[33] Zhang, Y., Meratnia, N., & Havinga, P. (2010). Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials, 12*(2), 159–170.
[34] Pandiri, L. Data-Driven Insights into Consumer Behavior for Bundled Insurance Offerings Using Big Data Analytics.
[35] Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks, 54*(15), 2787–2805.
[36] Bizer, C., & Cyganiak, R. (2009). Quality-driven information filtering using the WIQA policy framework. *Web Semantics: Science, Services and Agents on the World Wide Web, 7*(1), 1–10.
[37] Sriram, H. K., ADUSUPALLI, B., Singreddy, S., & Malempati, M. (2021). Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks. Murali, Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks (December 27, 2021).
[38] Bose, R. P. J. C., & Van der Aalst, W. M. P. (2009). Context aware trace clustering: Towards improving process mining results. *Proceedings of the SIAM International Conference on Data Mining*, 401–412.
[39] Pamisetty, V. (2021). Big Data and Predictive Analytics in Government Finance: Transforming Fraud Detection and Fiscal Oversight. Available at SSRN 5276847.
[40] Ni, K., Ramanathan, N., Chehade,M., Balzano,L., Nair,S., Zahedi,S., Kohler,E., Pottie,G., Hansen,M., & Srivastava,M. (2009). Sensor network data fault types. *ACM Transactions on Sensor Networks, 5*(3), 1–29.
[41] Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM, 51*(1), 107–113.
[42] Chava, K., Chakilam, C., & Recharla, M. (2021). Machine Learning Models for Early Disease Detection: A Big Data Approach to Personalized Healthcare. International Journal of Engineering and Computer Science, 10(12), 25709-25730.
[43] Olston, C., Reed, B., Srivastava, U., Kumar, R., & Tomkins, A. (2008). Pig Latin: A not-so-foreign language for data processing. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1099–1110.
[44] Meda, R. (2020). Real-Time Data Pipelines for Demand Forecasting in Retail Paint Distribution Networks. Global Research Development (GRD) ISSN, 2455-5703.
[45] Nandan, B. P. Data Analytics-Driven Approaches to Yield Prediction in Semiconductor Manufacturing. International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE), DOI, 10.
[46] Kolla, S. K. (2021). Designing Scalable Healthcare Data Pipelines for Multi-Hospital Networks. World Journal of Clinical Medicine Research, 1(1), 1-14.
[47] Stonebraker, M., Çetintemel, U., & Zdonik, S. (2005). The 8 requirements of real-time stream processing. *ACM SIGMOD Record, 34*(4), 42–47.
[48] Mangala, N. (2021). Optimizing Large-Scale ETL Pipelines Using Medallion Architecture on Azure Data Lake. Journal of Artificial Intelligence and Big Data, 1(1), 1-20.

[49] Abadi, D. J., Ahmad, Y., Balazinska, M., Çetintemel, U., Cherniack, M., Hwang, J. H., Lindner, W., Maskey, A., Rasin, A., Ryvkina, E., Tatbul, N., Xing, Y., & Zdonik, S. (2005). The design of the Borealis stream processing engine. *Proceedings of the Conference on Innovative Data Systems Research*, 277–289.
[50] Davuluri, P. N. (2020). Improving Data Quality and Lineage in Regulated Financial Data Platforms. Finance and Economics, 1(1), 1-14.
[51] Simmhan, Y. L., Plale, B., & Gannon, D. (2005). A survey of data provenance in e-science. *ACM SIGMOD Record, 34*(3), 31–36.
[52] Ranjith Kumar Peddi (2021). Optimizing Case Management Workflows in Global Data Center Colocation Services. Universal Journal of Computer Sciences and Communications, 1(1), 1-21. https://doi.org/10.31586/ujscs.2021.1380
[53] Gottimukkala, V. R. R. (2021). Digital Signal Processing Challenges in Financial Messaging Systems: Case Studies in High-Volume SWIFT Flows.
[54] Pamisetty, A. (2019). Big Data Engineering for Real-Time Inventory Optimization in Wholesale Distribution Networks. Available at SSRN 5267328.
[55] Pandiri, L. (2021). Machine Learning Approaches in Pricing and Claims Optimization for Recreational Vehicle Insurance. Journal of International Crisis and Risk Communication Research, 194-214.
[56] Recharla, M. (2020). Targeted Gene Therapy for Spinal Muscular Atrophy: Advances in Delivery Mechanisms and Clinical Outcomes. International Journal of Science and Research (IJSR), 9(12), 1921-1934.
[57] Nandan, B. P. (2021). Enhancing Chip Performance Through Predictive Analytics and Automated Design Verification. Journal of International Crisis and Risk Communication Research, 265-285.
[58] Singireddy, S., & Adusupalli, B. (2019). Cloud Security Challenges in Modernizing Insurance Operations with Multi-Tenant Architectures. International Journal of Engineering and Computer Science, 8, 12.
[59] Pandiri, L., Singireddy, S., & Adusupalli, B. (2020). Digital Transformation of Underwriting Processes through Automation and Data Integration. Global Research Development (GRD) ISSN, 2455-5703.
[60] Kummari, D. N. (2021). A Framework for Risk-Based Auditing in Intelligent Manufacturing Infrastructures. International Journal on Recent and Innovation Trends in Computing and Communication, 9(12), 245-262.
[61] Meda, R. (2020). Data Engineering Architectures for Real-Time Quality Monitoring in Paint Production Lines. International Journal Of Engineering And Computer Science, 9(12).
[62] Inala, R. Designing Scalable Technology Architectures for Customer Data in Group Insurance and Investment Platforms.
[63] Aitha, A. R. (2021). Optimizing Data Warehousing for Large Scale Policy Management Using Advanced ETL Frameworks.
[64] Valiki, D., & Kummari, D. N. (2021). Rule-Based Decision Systems for the Automation of Audit Sampling. International Journal of Emerging Trends in Computer Science and Information Technology, 2(4), 105-114.