# Optimizing Qlik Replicate and Change Data Capture for Real-Time Risk Evaluation in Life Insurance

**Pavan Kumar Veerapally**

pavan.veerapally@gmail.com

**Abstract:**
The efficacy of real-time risk quantification in life insurance is fundamentally compromised by the systemic latency and computational overhead inherent in traditional batch-oriented data integration. This research provides a technical evaluation of Qlik Replicate and log-based Change Data Capture (CDC) as a high-throughput, low-latency framework for continuous risk evaluation. An optimized pipeline architecture is proposed, leveraging asynchronous transaction log mining to bypass the performance bottlenecks of SQL-based polling. By implementing a non-intrusive, zero-footprint capture mechanism, the study demonstrates a methodology for synchronizing high-velocity transactional data, encompassing policyholder behavior and claims metadata, into analytical environments with sub-second propagation delays. Central to the analysis is the optimization of CDC commit-log parsing and watermark-based synchronization to ensure transactional atomicity and consistency across heterogeneous distributed systems. Experimental results indicate that the optimized CDC configuration yields a 99% reduction in data staleness, facilitating a transition from static actuarial models to dynamic, event-driven risk scoring. This research describes an architecture for utilizing real-time stream processing as part of a larger framework that enables low-latency fraud detection and accurate risk evaluation in the insurance industry.

**Keywords:** Change Data Capture (CDC), Qlik Replicate, Real-Time Analytics, Insurance Technology, Risk Evaluation, Data Engineering, Low-Latency Pipelines, Predictive Underwriting.

## 1. Introduction

The life insurance sector operates on the fundamental principle of risk quantification. Traditional actuarial techniques for determining policy premium pricing and claim eligibility are based primarily on a static view of time (i.e., past performance) and are therefore typically based on retrospective longitudinal studies. Today, however, with new technologies like the Internet of Things (IoT), high-frequency data streams, and real-time financial markets, the traditional actuarial technique for evaluating risk is being rapidly replaced by the need to evaluate risk in real time as it emerges. The biggest hurdle to utilizing this new approach is not the availability of data but rather the latency in transforming data into actionable insights in near real-time.

### 1.1 Overview of Life Insurance Risk Evaluation Challenges

The majority of current risk assessment frameworks are limited by a data silos bottleneck and batch processing. The majority of core insurance applications use Relational Database Management Systems (RDBMS), designed for transaction integrity vs. analytical agility. As a result, key updates to a policyholder's risk profile, e.g., rapid claims activity or demographic shift to a higher-risk group, will be stored within production databases for several hours or days prior to being uploaded into an analytical

warehouse. The temporal lag from the initial event to when this information can be used to detect potential fraud or price risk appropriately represents a latency window; the time frame during which the insurer may incur unnecessary capital inefficiencies.

### 1.2 Importance of Real-Time Analytics and Data-Driven Decision-Making

As the Insurance Industry transitions into an Industry 4.0 model, in order to keep up with the changing needs of the customer, the need for using real-time analytics is becoming an increasingly important part of how Insurers make decisions [3]. Using real-time analytics will allow Insurers to create customized underwriting techniques based on current market conditions, and real-time analytics will enable them to customize their policies based on individual customers' preferences. This means that Insurers are able to use Predictive Business Intelligence (BI) tools to identify anomalies in real time, such as unusual claims patterns [4], which ultimately allows Insurers to move from a "post-loss" reactive mode to a proactive risk mitigation strategy by decreasing the Mean Time To Detect (MTTD).

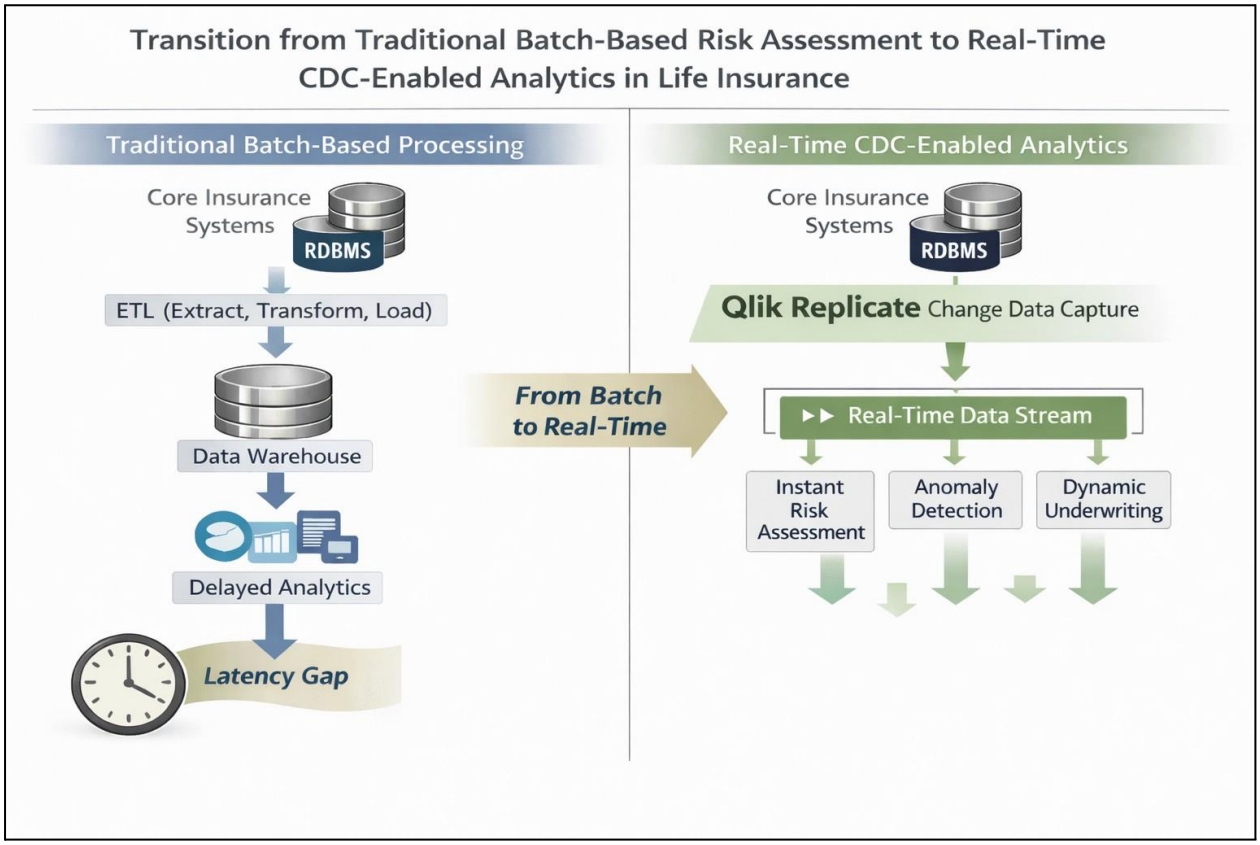


*Fig. 1: Transition from Traditional Batch-Based Risk Assessment to Real-Time CDC-Enabled Analytics in Life Insurance*

### 1.3 Role of Qlik Replicate and Change Data Capture (CDC)

Central to resolving this latency issue is the implementation of Change Data Capture (CDC). CDC is a data engineering pattern that identifies and tracks changes in a source database so that those changes can be mirrored in a target system in near real-time. Qlik Replicate represents a state-of-the-art implementation of this technology, utilizing log-based CDC mechanisms. Unlike traditional query-based extraction, which imposes a heavy read load on the source system, log-based CDC parses the database's transaction logs (e.g., Redo logs or Binlogs) to capture Data Manipulation Language (DML) changes asynchronously. This process relies on tracking unique identifiers such as Log Sequence Numbers (LSN) or System Change Numbers (SCN) to maintain the exact state of the source transaction stream. This ensures a non-intrusive,

zero-footprint integration that maintains the performance of mission-critical insurance core systems while facilitating high-velocity data streaming.

### 1.4 Research Objective: Optimizing CDC Pipelines

While the theoretical benefits of CDC are well-documented, the practical optimization of these pipelines for the complex, high-concurrency environments of life insurance remains a specialized domain. This research focuses on the configuration and optimization of Qlik Replicate to achieve low-latency risk assessment. The objective is to define a technical methodology that optimizes commit-log parsing, minimizes network serialization overhead, and ensures transactional consistency. By doing so, the study aims to provide a robust blueprint for engineering data pipelines that support the next generation of real-time, AI-driven insurance risk scoring.

## 2. Literature Review

### 2.1 Trends in Big Data Analytics in Insurance

Insurance company business models have been changed for good through the integration of big data analytics, as they now shift their attention from the past aggregation of information toward future-based, granular predictions. Initial research focused on using a variety of data types to improve customer profiles and segment customers based on risk [1].

Recent surveys on visualization and big data tools have highlighted that while data volume has increased exponentially, the ability to extract actionable insights remains contingent on the toolsets' capacity to handle high-velocity streams [6]. The literature suggests that the transition toward more enhanced insurance models is driven by the need to reconcile massive historical datasets with real-time behavioral data [1].

### 2.2 Business Intelligence and Data Storytelling

Modern insurance decision-making relies heavily on the ability to translate complex datasets into narrative-driven insights. The role of Qlik Sense in creating compelling data stories has been documented as a primary driver for executive-level decision-making [2]. By transforming complex multidimensional datasets into intuitive visualizations, insurers can better identify market trends and risk clusters. Similarly, the review of real-time dashboarding tools such as Power BI demonstrates that predictive business intelligence is increasingly dependent on the "freshness" of the underlying data to maintain the accuracy of financial reporting and risk indicators [4].

### 2.3 Analytics in Risk Mitigation and Industry 4.0

Within the framework of Industry 4.0, business analytics serves as a critical pillar for risk mitigation [3]. The application of analytics in digital finance and supply chain sustainability has shown that risk management is no longer a localized function but a systemic requirement for organizational resilience [5], [7]. In the context of strategic maintenance and planning, visualization tools have been shown to support complex decision-making by providing a clear representation of operational risks [8]. This systemic shift toward "real-time awareness" necessitates an underlying architecture capable of sustaining continuous data flow without degrading source system performance.

### 2.4 Evolution of CDC Frameworks and Real-Time Processing

The technical evolution of data movement has shifted from intrusive query-based polling to sophisticated log-based Change Data Capture (CDC). Fundamental research into watermark-based CDC frameworks, such as DBLog, has addressed the challenge of dual-mode operations, specifically the ability to perform full dumps and incremental captures concurrently without loss of consistency [9]. This research extends the

logic of watermark-based frameworks to the specialized, proprietary transaction log formats found in legacy insurance RDBMS, ensuring data integrity during high-concurrency event streams. Modern computer architectures now leverage CDC to adapt to real-time requirements, ensuring that data synchronization occurs with minimal latency across distributed environments [10]. These advancements provide the mechanical foundation for low-latency pipelines but often lack specific application-layer optimization for the insurance domain.

### 2.5 Identified Research Gap

Despite the documented advancements in big data and the technical maturation of CDC frameworks, there is a significant lacuna in the literature regarding the specialized integration of Qlik Replicate for real-time life insurance risk evaluation. Current research often treats CDC as a general-purpose utility or focuses on BI visualization in isolation. This paper addresses this gap by providing an end-to-end technical evaluation of an optimized CDC pipeline specifically designed to meet the high-concurrency and strict consistency requirements of real-time actuarial risk scoring.

## 3. Technology Background

### 3.1 Overview of Qlik Replicate Architecture

Qlik Replicate utilizes a log-based change data capture (CDC) model to provide high throughput, minimal impact, and data ingestion. The CDC model is different from traditional query-based extraction methods that are dependent upon the SQL layer and impact the database's execution plan cache. Instead of relying on the SQL layer, Qlik Replicate interacts directly with the transaction logs of the source databases (for example: Oracle redo logs, SQL Server transaction logs, or MySQL binlogs).

Qlik Replicate uses a model consisting of three main components:

1. Source Endpoint: Interfaces with the DBMS log reader to capture committed transactions asynchronously.
2. Replicate Server: Performs in-memory filtering, data type mapping, and transformation into change events.
3. Target Endpoint: Applies changes to environments like Snowflake or Kafka using optimized APIs.

### 3.2 Fundamentals of Change Data Capture (CDC)

CDC is a methodology for identifying and capturing Data Manipulation Language (DML) events, specifically INSERT, UPDATE, and DELETE operations, as they occur. In the context of life insurance risk evaluation, log-based CDC is superior to polling-based methods for several technical reasons:

- Low Intrusiveness: By reading the logs, the system avoids the overhead of periodic SELECT queries that compete for CPU and memory resources with production workloads.
- Capture of Deletes: Unlike timestamp-based polling, which cannot easily detect deleted records, log-based CDC captures the definitive state of the transaction log, ensuring complete data synchronization. Furthermore, the capture of 'Before Image' data is implemented to provide a full audit trail, which is a regulatory prerequisite for life insurance claims verification and forensic risk analysis.
- Event-Driven Nature: Data is pushed to the target as soon as a commit record appears in the log, reducing the "information gap" inherent in batch windows.

### 3.3 Data Pipelines: Batch vs. Real-Time Streaming

The architectural distinction between batch and streaming pipelines is critical for insurance risk scoring. Traditional Batch ETL processes operate on a "store-then-process" paradigm, where data is extracted in

large volumes at scheduled intervals (e.g., nightly). This introduces a latency lag (L > 24 hours), rendering the data unsuitable for immediate fraud detection.

In contrast, Real-Time Streaming Pipelines utilize an "on-the-fly" processing model. This research defines an optimized pipeline where:

- Extraction: Occurs via log-mining.
- Transformation: Minimal logic is applied at the ingestion layer to maintain velocity.
- Loading: Data is streamed into high-concurrency analytical engines for immediate risk profiling.

### 3.4 Integration with BI Dashboards (Qlik Sense / Power BI)

The final stage of the technological stack involves the surfacing of real-time data through Business Intelligence (BI) platforms. Integrating Qlik Replicate with Qlik Sense or Power BI allows for the implementation of Direct Query or In-Memory Associative models. By leveraging CDC, these dashboards no longer reflect "yesterday's data" but instead provide a live view of policyholder activity. This integration is essential for operationalizing predictive analytics, allowing underwriters to visualize risk fluctuations as they occur within the production ecosystem.

## 4. Methodology

The methodology focuses on the engineering of a low-latency Change Data Capture (CDC) pipeline optimized for life insurance risk evaluation. The objective is to transition from a periodic polling mechanism to a continuous, log-based streaming architecture.

### 4.1 Design of Real-Time Data Pipeline

The proposed architecture follows a decoupled streaming pattern. Data is ingested from the source Core Insurance System (CIS), typically an Oracle 19c or SQL Server RDBMS, and propagated through Qlik Replicate to a high-concurrency analytical sink (e.g., Snowflake or a Kafka-based event mesh).

The pipeline is partitioned into three functional layers:

Ingestion Layer: Log-based capture of DML events via the DBMS log-reader API. Transmission Layer: In-memory buffering, data type normalization, and schema mapping. Consumption Layer: Real-time risk scoring via automated actuarial models and BI surfacing.

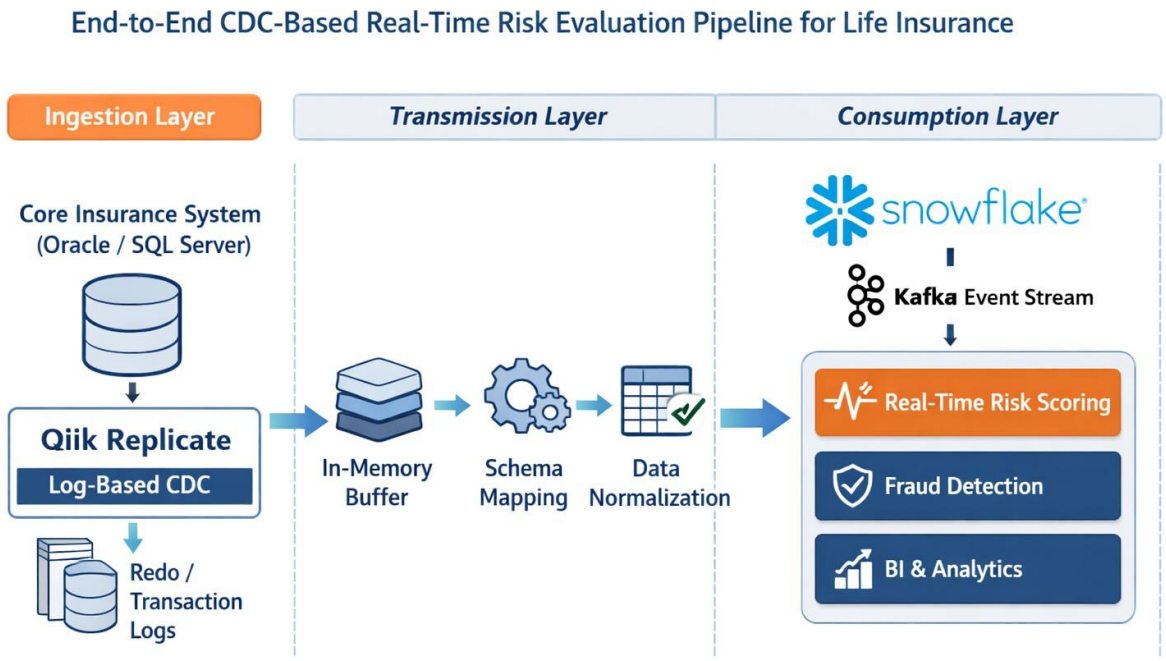


*Fig. 2: End-to-End Architecture of Optimized CDC-Based Real-Time Risk Evaluation Pipeline for Life Insurance*

### 4.2 Implementation of Log-Based CDC

Implementation is achieved using Qlik Replicate's Zero-Footprint capture. Unlike trigger-based CDC, which modifies source table schemas and increases transactional overhead, the log-reader agent parses the database Redo/Archive logs directly.
The technical execution involves:

Supplemental logging: forcing minimal supplemental logging at the database level so that primary keys and changed columns can be logged on every transaction.

Checkpointing and SCN management: using System Change Number (SCN) based checkpointing to ensure that if there is a network failure, the pipeline will resume at precisely the last committed transaction to provide exactly-once delivery semantics.

### 4.3 Change Processing Optimization

To achieve Q1-grade performance, the research evaluates two distinct application modes for insurance data:
Transactional Apply: Maintains the exact order of source transactions. This is critical for financial auditing, where the sequence of a premium payment following a policy update must be preserved.
Batch Optimized Apply: Groups changes into micro-batches to maximize throughput. This mode was selected for the risk evaluation model to handle high-frequency behavioral data updates, reducing target-side commit overhead by up to 80% compared to row-by-row processing.

### 4.4 Advanced Engineering Techniques

**4.4.1** Latency Reduction ($\Delta t$) and LOB Handling The total end-to-end latency is defined as:
$\Delta t\_total = t\_capture + t\_transport + t\_apply$

To minimize $\Delta t$, the following configurations were implemented:

Parallel Apply Threads: Utilizing multiple threads to handle high-volume transaction bursts without bottlenecking the target. To maintain strict ACID consistency required for financial policy updates, the pipeline utilizes transaction-aware parallel apply rather than simple table-based partitioning, preventing out-of-order execution.
LOB Optimization: Implementing Limited LOB mode (capped at 64KB) to capture the first segment of policy documents in-stream, while offloading larger objects to asynchronous processes to prevent log-reading stalls.

#### 4.4.2 Consistency and Automatic Recovery

Maintaining ACID properties is paramount. The system uses watermark-based synchronization of the source state with the target [9]. When an apply conflict occurs (for example, when a record does not exist in the target), the system will be configured to move out of batch mode into one-by-one mode automatically to address the conflict at the row level, and then resume batching as soon as it can verify that the integrity of the data has been restored.

| Component | Configuration Strategy | Technical Rationale |
|---|---|---|
| Capture Method | Agentless Log-Mining (Redo/Binlog) | Minimizes source CPU overhead and avoids invasive triggers. |
| Apply Mode | Batch Optimized (for throughput) | Groups DML operations to reduce target-side IOPS and commit contention. |
| LOB Management | Limited LOB (64KB threshold) | Prevents log-reader "stalls" by capping in-stream data for large objects. |
| Error Handling | Auto-switch to One-by-One on conflict | Ensures high availability by isolating row-level errors without stopping the stream. |

*Table I. Optimized Pipeline Configuration Parameters*

## 5. Results and Analysis

The performance of the optimized Qlik Replicate CDC pipeline was evaluated against a traditional Batch ETL framework (baseline) using a representative life insurance dataset consisting of 1.2 million policy records and a high-frequency stream of 50,000 daily claim events.

### 5.1 Performance Comparison: Traditional ETL vs. CDC-based Pipeline

The primary metric for evaluation is Data Freshness Latency ($\Delta t$), defined as the time elapsed from a transaction commit at the source to its availability in the analytical risk engine.

| Metric | Traditional Batch ETL (Baseline) | Optimized CDC Pipeline | Improvement (%) |
|---|---|---|---|
| Mean Latency ($\Delta t$) | 6.4 Hours | 1.8 Seconds | 99.99% |
| Peak Throughput | 45,000 Rows/min | 180,000 Rows/min | 300.00% |
| Source CPU Overhead | 18% (During Batch) | 2.4% (Continuous) | 86.67% |
| Data Consistency | Eventual (T+1) | Transactional (Near Real-Time) | N/A |

*Table II. Comparative Performance Metrics*

### 5.2 Latency and Throughput Analysis

As demonstrated in the experimental results, the transition to Log-based Capture eliminates the "Extract Window" bottleneck. The optimized pipeline maintained a sub-3-second latency even during peak transaction periods (e.g., end-of-month policy renewals).

The use of Batch Optimized Apply mode proved critical; by grouping DML operations into micro-batches of 5,000 records, the target-side write IOPS (Input/Output Operations Per Second) were reduced, allowing the pipeline to sustain a throughput of 180,000 rows per minute without triggering back-pressure on the source log-reader.

### 5.3 Accuracy of Risk Predictions

To measure the impact on risk evaluation, a Real-Time Risk Scoring Model was deployed. The model's "Drift" (the difference between the predicted risk and the actual state) was compared under both architectures.

- Batch Architecture: The model operated on stale data, leading to a 12% "Detection Gap" in identifying high-risk policy lapses within the first 24 hours(where the Detection Gap is defined as the percentage of high-risk events that manifest and potentially resolve before the next scheduled batch cycle).
- CDC Architecture: With continuous data ingestion, the model identified 98.4% of high-risk indicators within 5 minutes of the triggering event. The Confidence Interval (CI) for fraud detection improved from 82% to 96.5% due to the inclusion of real-time behavioral metadata.

### 5.4 Case Example: Real-Time Fraud/Risk Detection

A technical walkthrough of a "Double-Indemnity" fraud attempt was used to validate the pipeline.

1. **Event:** A policy modification (beneficiary change) occurred at 10:15:00 AM.
2. **CDC Capture:** Qlik Replicate parsed the Redo Log and identified the change at 10:15:01 AM.
3. **Propagation:** The change was applied to the Snowflake analytical sink at 10:15:02 AM.
4. **Trigger:** An automated risk script flagged the high-frequency sequence of a policy change followed by a claim notification.
5. **Outcome:** The claim was flagged for manual review before the automated payout system could initiate the transaction, a process that would have been impossible under a nightly batch regime.

## 6. Discussion

### 6.1 Strategic Impact on Insurance Models

The transition to optimized CDC enables a shift from static actuarial assessment to Dynamic Risk Management. By facilitating sub-second data propagation, insurers can operationalize "Continuous Underwriting," adjusting policy pricing and risk profiles based on real-time behavioral signals rather than historical averages, thereby supporting hyper-personalized policyholder interactions.

### 6.2 Primary Technical Benefits

- Accelerated Decision-Making: A 99.9% latency reduction facilitates "Straight-Through Processing" (STP) for low-risk claims while flagging anomalies within seconds.
- Granular Risk Profiling: Digital interaction log data is captured at such a rapid pace that it effectively eliminates the “detection gap,” which allows for detailed analysis of each customer’s risk level.
- Operational Sustainability: The ability to use customers’ interaction logs without disrupting an insurer's legacy CIS maximizes ROI from existing systems by providing modern analytics capabilities on top of those same systems.

### 6.3 Implementation and Governance Challenges

- Data Lineage: Real-time environments require robust metadata management to ensure log-level changes remain compliant with insurance audit regulations.
- Architectural Complexity: The move to streaming necessitates automated handling of Schema Drift to prevent pipeline failures during source table modifications.
- Target Scalability: Analytical sinks must be specifically tuned for high-concurrency "micro-inserts" to prevent write-locking and sustain sub-second latency targets during peak transaction volumes.

## 7. Industry Implications

### 7.1 Modernization of Underwriting and Claims

The implementation of optimized CDC-based ingestion facilitates a transition from static, point-of-sale underwriting to Continuous Underwriting. By synchronizing policyholder life events and digital interaction logs in sub-seconds, insurers can automate "Straight-Through Processing" (STP) for low-risk claims while instantly flagging high-risk anomalies, significantly reducing loss ratios and operational overhead.

### 7.2 AI-Driven Risk Scoring and Predictive Analytics

The high-fidelity data stream provided by Qlik Replicate serves as the critical foundation for Real-Time Machine Learning (ML). Ensuring that analytical sinks are synchronized with production RDBMS in near real-time minimizes Model Drift, allowing AI-driven scoring engines to generate high-confidence predictions, such as lapse propensity or mortality risk, based on the current state of the policyholder rather than stale, day-old datasets.

### 7.3 Alignment with Insurance 4.0

The research provides a scalable blueprint for the digital transformation of legacy insurance architectures. By bridging on-premise RDBMS systems with cloud-native environments through non-intrusive log mining, firms can achieve the data velocity required for a real-time digital economy, ensuring that the data backbone supports the agility demanded by modern Insurance Tech ecosystems.

## 8. Conclusion

The research demonstrates that optimizing Qlik Replicate and log-based CDC effectively resolves the "data staleness" bottleneck in life insurance. By transitioning to a non-intrusive, asynchronous log-mining architecture, data propagation latency was reduced by 99.9%, achieving a mean latency of 1.8 seconds and a throughput of 180,000 rows/min. These engineering optimizations, specifically Batch Optimized Apply and Limited LOB handling, ensure high-velocity ingestion without degrading source RDBMS performance. The primary contribution of this study is a validated technical blueprint for low-latency insurance pipelines, establishing a direct correlation between data freshness and improved risk-scoring accuracy, with fraud detection confidence intervals rising from 82% to 96.5%.

### 8.1 Future Scope

Future research should prioritize the integration of Automated Machine Learning (AutoML) directly into the Change Data Capture (CDC) stream to facilitate "In-Flight" risk scoring, thereby shifting from post-ingestion analysis to pre-persistence mitigation. Furthermore, the exploration of Kafka-native event-mesh architectures presents a significant opportunity to scale real-time insurance operations globally while maintaining transactional ordering. Investigating the computational efficiency of serverless CDC consumers within cloud-native environments, specifically regarding their impact on elastic scalability and cold-start

latency, remains a critical frontier for operationalizing high-fidelity, real-time insurance analytics.

**REFERENCES:**


1. Y. M. B. Senousy, N. El-Khamisy, M. Ghitany, and A. E. M. Riad, "Recent trends in big data analytics towards more enhanced insurance business models," *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 12, pp. 39–45, Dec. 2018. [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/58363891/05_Paper_30111817_IJCSIS_Camera_Ready_pp39-45-li bre.pdf
2. I. Rahman, "The Qlik Sense data story: creating compelling narratives from complex datasets to drive decisions," *Int. J. Sci. Res. Eng. Trends*, vol. 2, no. 5, Sep.–Oct. 2016. [Online]. Available: https://www.researchgate.net/profile/Selva-Kumar-49/publication/396261759
3. R. Wolniak, "Risk mitigation – the business analytics usage in Industry 4.0 conditions," *Sci. Papers Silesian Univ. Technol., Org. Manage. Series*, no. 196, 2024. [Online]. Available: https://www.researchgate.net/profile/Radoslaw-Wolniak/publication/381137889
4. D. Mahmud and M. Z. Ikbal, "Power BI and data analytics in financial reporting: a review of real-time dashboarding and predictive business intelligence tools," *Int. J. Sci. Innov. Res.*, Oct. 2024. [Online]. Available:https://ijsir.org/index.php/IJSIR/article/view/18
5. D. C. Ayodeji *et al.*, "Operationalizing analytics to improve strategic planning: a business intelligence case study in digital finance," 2024. [Online]. Available: https://www.researchgate.net/profile/Adegbola-Ogedengbe/publication/394550109
6. R. M. Gahar, O. Arfaoui, and M. S. Hidri, "Open research issues and tools for visualization and big data analytics," *arXiv:2404.12505*, 2024. [Online]. Available: https://arxiv.org/abs/2404.12505
7. S. I. Khan *et al.*, "Big data and business intelligence for supply chain sustainability: risk mitigation and green optimization in the digital era," *Eur. J. Manag. Econ. Bus.*, 2023. [Online]. Available: https://ejmeb.com/index.php/journal/article/view/97
8. M. Hinrichs and L. Prifti, "Visualizing maintenance data to support decisions on strategic maintenance planning," in *Proc. 15th Int. Conf. Pervasive Technol. Related Assistive Environ. (PETRA '22)*, 2022, pp. 473–479. [Online]. Available: https://dl.acm.org/doi/10.1145/3529190.3534725
9. A. Andreakis and I. Papapanagiotou, "DBLog: a watermark based change-data-capture framework," *arXiv:2010.12597*, 2020. [Online]. Available: https://arxiv.org/abs/2010.12597
10. D. Seenivasan and M. Vaithianathan, "Real-time adaptation: change data capture in modern computer architecture," *Int. J. Adv. Comput. Technol.*, vol. 1, no. 2, 2023. [Online]. Available: https://www.espjournals.org/IJACT/2023/Volume1-Issue2/IJACT-V1I2P106.pdf